



**ФЕДЕРАЛЬНАЯ СЛУЖБА  
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ**

**(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ**

(21)(22) Заявка: **2009125945/07, 06.07.2009**

(24) Дата начала отсчета срока действия патента:  
**22.04.2004**

Приоритет(ы):

(30) Конвенционный приоритет:  
**25.04.2003 JP 2003-121595**

Номер и дата приоритета первоначальной заявки,  
из которой данная заявка выделена:  
**2004137807 22.04.2004**

(43) Дата публикации заявки: **20.01.2011** Бюл. № 2

(45) Опубликовано: **20.12.2011** Бюл. № 35

(56) Список документов, цитированных в отчете о  
поиске: **EP 0847198 A1, 10.06.1998. RU 2163057 C2,  
10.02.2001. EP 0847200 A1, 10.06.1998.  
EP 1198132 A1, 17.04.2002.**

Адрес для переписки:

**103735, Москва, ул.Ильинка, 5/2, ООО  
"Союзпатент", пат.пов. С.В.Истомину**

(72) Автор(ы):

**ХАМАДА Тосия (JP),  
КАТО Мотоки (JP)**

(73) Патентообладатель(и):

**СОНИ КОРПОРЕЙШН (JP)**

**(54) УСТРОЙСТВО ВОСПРОИЗВЕДЕНИЯ, СПОСОБ ВОСПРОИЗВЕДЕНИЯ, ПРОГРАММА ДЛЯ  
ВОСПРОИЗВЕДЕНИЯ И НОСИТЕЛЬ ЗАПИСИ**

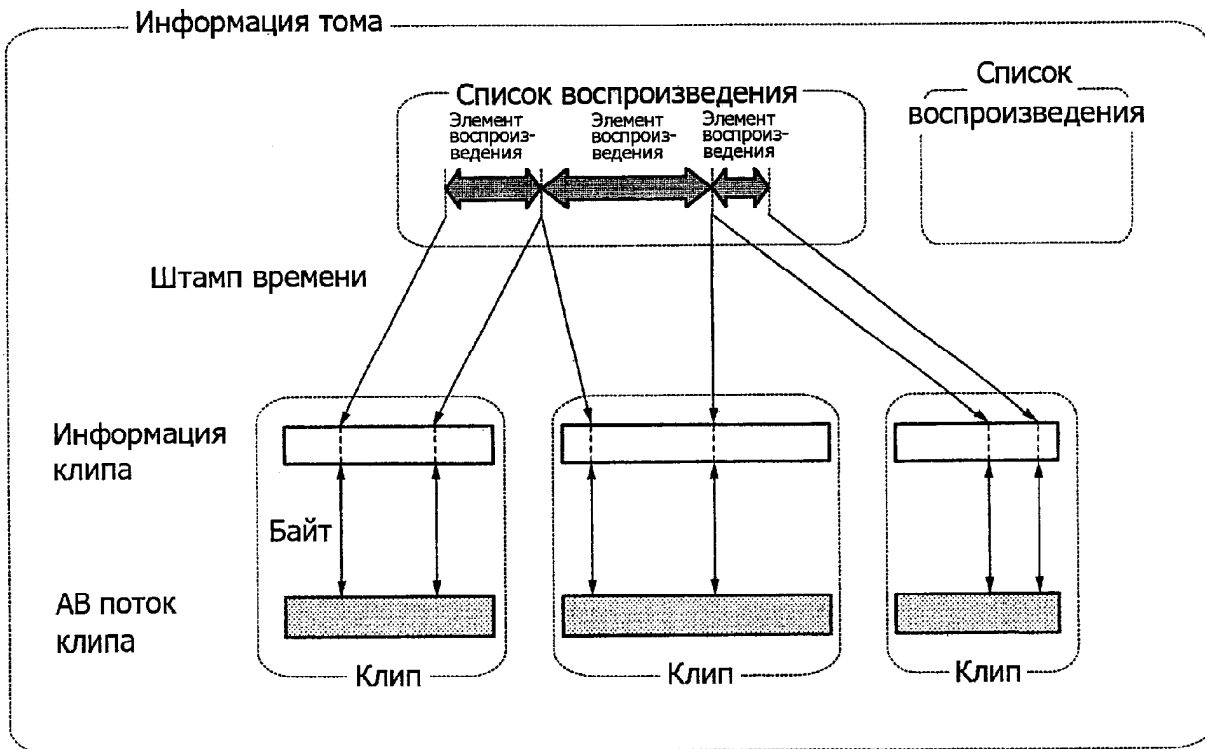
(57) Реферат:

Изобретение относится к устройствам воспроизведения. Техническим результатом является отображение элемента воспроизведения с множеством углов обзора. Результат достигается тем, что описан флаг, который представляет, действительно ли элемент воспроизведения имеет структуру множества углов обзора, в которой элемент воспроизведения воспроизводят с множеством углов обзора. Кроме того, описан флаг, который представляет, действительно ли начало каждого

модуля декодирования представляет собой точку возможности переключения угла. В бесстыковой структуре множества углов текущий угол можно переключать без необходимости увеличения количества модулей чередования. В небесстыковой структуре множества углов, с использованием флага, в заданной области на стороне окончания каждого угла, запрещено переключать текущий угол. При этом, когда элемент воспроизведения выходит из множества углов, предотвращается разрыв воспроизведения. 4 н. и 8 з.п. ф-лы, 9б ил.

RU 2 437 243 C2

RU 2 437 243 C2



Фиг. 1

RU 2 4 3 7 2 4 3 C 2

RU 2 4 3 7 2 4 3 C 2



FEDERAL SERVICE  
FOR INTELLECTUAL PROPERTY

(12) **ABSTRACT OF INVENTION**

(21)(22) Application: **2009125945/07, 06.07.2009**

(24) Effective date for property rights:  
**22.04.2004**

Priority:

(30) Priority:  
**25.04.2003 JP 2003-121595**  
**2004137807 22.04.2004**

(43) Application published: **20.01.2011 Bull. 2**

(45) Date of publication: **20.12.2011 Bull. 35**

Mail address:

**103735, Moskva, ul.II'inka, 5/2, OOO**  
**"Sojuzpatent", pat.pov. S.V.Istominu**

(72) Inventor(s):

**KhAMADA Tosija (JP),**  
**KATO Motoki (JP)**

(73) Proprietor(s):

**SONI KORPOREJShN (JP)**

(54) **REPRODUCTION DEVICE, METHOD OF REPRODUCTION, PROGRAMME FOR REPRODUCTION AND RECORDING MEDIUM**

(57) Abstract:

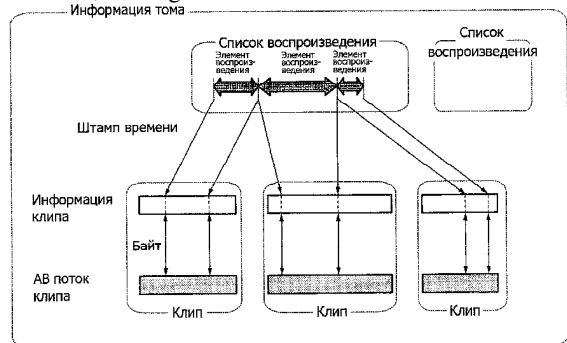
FIELD: information technologies.

SUBSTANCE: flag is described, which represents whether a reproduction element actually comprises a structure of multiple view angles, in which a reproduction element is reproduced with multiple view angles. Besides, a flag is described, which represents whether start of each decoding module is actually a point of angle switching possibility. In a jointless structure of multiple angles a current angle may be switched without the necessity to increase quantity of alternating modules. In a non-jointless structure of multiple angles, with application of a flag, in the specified area at the side of each angle ending, it is prohibited to switch the current angle. At the same time, when the

reproduction element comes out of multiple angles, reproduction break is prevented.

EFFECT: displaying a reproduction element with multiple view angles.

12 cl, 96 dwg



Фиг. 1

RU 2 4 3 7 2 4 3 C 2

RU 2 4 3 7 2 4 3 C 2

Текст описания приведен в факсимильном виде.

### **Область техники, к которой относится изобретение**

5 Настоящее изобретение относится к устройству воспроизведения, способу воспроизведения, программе для воспроизведения, и носителю записи, которые позволяют пользователю, в интерактивном режиме использовать программу, записанную на носитель записи большой емкости, такой как диск типа "голубой луч" (blu-ray).

10

### **Уровень техники**

В последние годы, в качестве стандарта для носителя записи типа съемного диска с возможностью записи для устройства записи и воспроизведения, был предложен стандарт 15 диска типа "голубой луч". Стандарт диска типа "голубой луч" определяет диск, который представляет собой носитель записи диаметром 12 см со слоем покрытия толщиной 0,1 мм. Стандарт диска типа "голубой луч" определяет использование сине-фиолетового лазера с длиной волны 405 нм, и объектива с числовой апертурой 0,85. Стандарт диска 20 типа "голубой луч" позволяет обеспечить максимальную емкость записи 27 Гб (Гбайт). На такой диск может быть записана двухчасовая программа передач японского телевидения BS высокой четкости или дольше, без ухудшения качества изображения.

25

В качестве источников (источников поставки) АВ (AV, аудио/видео) сигналов, записываемых на оптический диск с возможностью записи, используется аналоговый сигнал, например, сигнал обычного аналогового телевидения и цифровой сигнал, 30 например, сигнал цифрового телевидения, такого как сигнал цифровой передачи телевидения BS. Стандарт диска типа "голубой луч" определяет способ записи таких АВ сигналов передачи.

35

В качестве стандарта, производного от используемого в настоящее время стандарта диска типа "голубой луч", разрабатывается носитель записи, предназначенный только для воспроизведения, на котором может быть заранее записан кинофильм, музыка или тому 40 подобное. В настоящее время широкое распространение получил носитель записи в форме диска типа DVD (цифровой универсальный диск), на котором заранее записаны кинофильм или музыка. Оптический диск, предназначенный только для воспроизведения, в соответствии со стандартом диска типа "голубой луч" в значительной степени отличается и превосходит по своим характеристикам обычный диск DVD, благодаря 45 большой емкости записи и высокой скорости передачи данных, что позволяют обеспечить

50

запись изображения телевидения высокой четкости с высоким качеством в течение двух часов или дольше.

5 Применяемый в настоящее время стандарт диска типа "голубой луч" не определяет ни способ отображения списка видеосодержания диска на экране, ни функцию интерфейса  
пользователя, который позволяет пользователю перемещать курсор по списку для выбора  
10 видеосодержания, которое он или она хотел бы воспроизвести из списка. Эти функции  
выполняются с помощью основного блока устройства записи и воспроизведения, которое  
осуществляет запись и воспроизведение видеосодержания на диск и с диска типа  
"голубой луч". При этом даже если видеосодержание воспроизводится с одного и того же  
15 носителя записи, вид экрана при отображении списка содержания, а также интерфейс  
пользователя, зависят от используемого устройства записи и воспроизведения. В  
результате для пользователя затрудняется использование диска типа "голубой луч". Таким  
образом, при использовании диска, предназначенного только для воспроизведения,  
20 необходимо обеспечить возможность отображения на экране меню и т.д., разработанного  
производителем диска (содержания), не зависимо от устройства воспроизведения.

Функция отображения экрана выбора в виде множества текстовых блоков при  
25 воспроизведении видеосодержания обычно называется интерактивной функцией. Для  
выполнения такой интерактивной функции необходимо, чтобы производитель диска  
создал сценарий, в котором он или она определил бы порядок воспроизведения и разделы  
видеосодержания, описал этот сценарий с использованием языка программирования,  
30 языка сценариев, или подобного языка, и записал описанный сценарий на диске.  
Устройство воспроизведения затем считывает и выполняет этот сценарий. В результате,  
устройство воспроизведения воспроизводит видеосодержание и отображает экраны  
35 выбора, которые позволяют пользователю выбирать разделы видеосодержания,  
определенные производителем.

Существующий в настоящее время стандарт диска типа "голубой луч" (формат  
40 диска типа голубой луч с возможностью перезаписи, версия 1.0) не определяет ни способ  
составления экрана меню и экрана выбора разделов, которые определил производитель  
содержания, ни способ описания процесса ввода команд пользователя. В настоящее время  
трудно обеспечить воспроизведение видеосодержания с диска типа "голубой луч" в  
45 соответствии со сценарием, который определил производитель, при обеспечении  
совместимости, независимо от изготовителя и модели устройства воспроизведения.

50 Сторона производителя стремится обеспечить на диске, предназначенном только  
для воспроизведения, функцию множества углов обзора, которая позволяет пользователю  
выбирать один из углов обзора объекта, снятого множеством камер, что позволяет

пользователю наблюдать объект с точки обзора, которая ему больше всего нравится. Таким образом, необходимо обеспечить такую функцию.

5  
10  
15  
20  
25  
30  
35  
40  
45  
50  
Описанная выше интерактивная функция была ранее определена, например, в стандарте DVD (цифровой универсальный диск). При воспроизведении движущегося изображения с диска DVD можно вызывать экран меню, например, используя устройство дистанционного управления. Путем нажатия на кнопку, отображаемую на экране меню, пользователь может выполнять процесс изменения текущей воспроизводимой сцены. Кроме того, видеодиск DVD также имеет функцию множества углов обзора.

Интерфейс пользователя и интерактивная функция, которые были разработаны для обычных носителей записи типа дисков DVD, также желательны для носителей записи большой емкости, таких как диски типа "голубой луч", предназначенные только для воспроизведения.

### 20 **Сущность изобретения**

В соответствии с этим настоящее изобретение направлено на создание устройства воспроизведения, способа воспроизведения, программы для воспроизведения и носителя записи, которые позволяют обеспечить для пользователя интерфейс, обладающий высокой степенью гибкости, для носителя записи большой емкости.

Кроме того, в соответствии с другой целью настоящее изобретение направлено на создание устройства воспроизведения, способа воспроизведения, программы для воспроизведения и носителя записи, которые позволяют обеспечить расширенный интерфейс пользователя для носителя записи большой емкости.

Для решения указанной выше проблемы настоящее изобретение направлено на создание устройства воспроизведения, предназначенного для воспроизведения данных содержания, записанных на носитель записи в форме диска, причем устройство воспроизведения содержит: средство считывания, предназначенное для считывания с носителя записи видеопотока, первого флага, описанного для каждого воспроизводимого блока видеопотока, количества углов, которые содержит воспроизводимый блок, и информации о положении, которая представляет угловые положения в видеопотоке, причем первый флаг представляет, действительно ли воспроизводимый блок может воспроизводиться под множеством углов, средство воспроизведения, предназначенное для управления средством считывания для считывания видеопотока, который составляет углы, в соответствии с информацией о положении, и выполнено с возможностью изменения положения воспроизведения видеопотока в соответствии с информацией о положении так, что обеспечивается возможность переключения текущего угла, когда

первый флаг представляет, что воспроизводимый блок можно воспроизводить под множеством углов, в котором видеопоток был закодирован в модуле одного или множества кадров и записан на носитель записи, в котором средство считывания  
5 выполнено с возможностью считывания второго флага, описанного в модуле кодирования, с носителя записи, причем второй флаг представляет, действительно ли обеспечивается возможность переключения текущего угла в начале модуля кодирования, и в котором  
10 средство воспроизведения сконфигурировано с возможностью изменения положения воспроизведения видеопотока в положении в соответствии со вторым флагом.

Кроме того, настоящее изобретение направлено на создание способа воспроизведения, предназначенного для воспроизведения данных содержания, записанных на носитель записи в форме диска, причем способ воспроизведения содержит  
15 следующие этапы: считывание с носителя записи видеопотока, первого флага, описанного для каждого воспроизводимого блока видеопотока, количества углов, которые содержит воспроизводимый блок, и информации о положении, которая представляет угловые  
20 положения в видеопотоке, причем первый флаг представляет, действительно ли воспроизводимый блок может быть воспроизведен с множеством углов, и управление этапом считывания для считывания видеопотока, который составляет углы, в соответствии с информацией о положении, при этом изменяют положения воспроизведения видеопотока в соответствии с информацией о положении так, что  
25 обеспечивается возможность переключения текущего угла, когда первый флаг представляет, что воспроизводимый блок можно воспроизводить под множеством углов, в котором видеопоток был закодирован в модуле одного или множества кадров и записан на носитель записи, в котором дополнительно осуществляют считывание второго флага,  
30 описанного в модуле кодирования, с носителя записи, причем второй флаг представляет, действительно ли обеспечивается возможность переключения текущего угла в начале модуля кодирования, и изменяют положение воспроизведения видеопотока в положении в соответствии со вторым флагом.  
40

Кроме того, настоящее изобретение направлено на создание носителя записи в форме диска, на который была записана программа воспроизведения, которая может быть считана компьютерным устройством, причем программа воспроизведения обеспечивает  
45 возможность выполнения с помощью компьютерного устройства способа воспроизведения, предназначенного для воспроизведения данных содержания, записанных на носитель записи, причем способ воспроизведения содержит этапы: считывание с носителя записи видеопотока, первого флага, описанного для каждого  
50 воспроизводимого блока видеопотока, количества углов, которые содержит

воспроизводимый блок, и информации о положении, которая представляет угловые  
положения в видеопотоке, причем первый флаг представляет, действительно ли  
воспроизводимый блок может быть воспроизведен с множеством углов, управление  
5 этапом считывания для считывания видеопотока, который составляет углы, в  
соответствии с информацией о положении, при этом изменяют положения  
воспроизведения видеопотока в соответствии с информацией о положении так, что  
10 обеспечивается возможность переключения текущего угла, когда первый флаг  
представляет, что воспроизводимый блок можно воспроизводить под множеством углов, в  
котором видеопоток был закодирован в модуле одного или множества кадров и записан на  
носитель записи, в котором дополнительно осуществляют считывание второго флага,  
15 описанного в модуле кодирования, с носителя записи, причем второй флаг представляет,  
действительно ли обеспечивается возможность переключения текущего угла в начале  
модуля кодирования, и изменяют положение воспроизведения видеопотока в положении в  
20 соответствии со вторым флагом.

Кроме того, настоящее изобретение направлено на создание носителя записи в  
форме диска, на который были записаны данные содержания, в котором видеопоток,  
25 первый флаг, описанный для каждого воспроизводимого блока видеопотока, количество  
углов, которые содержит воспроизводимый блок, и информацию о положении, которая  
представляет угловые положения в видеопотоке, были записаны на носителе записи,  
причем первый флаг представляет, действительно ли воспроизводимый блок может быть  
30 воспроизведен с множеством углов, и в котором видеопоток, который составляет углы,  
считывают в соответствии с информацией о положении, при этом положение  
воспроизведения видеопотока можно изменять в соответствии с информацией о  
35 положении, когда первый флаг представляет, что воспроизводимый блок может быть  
воспроизведен с множеством углов, в котором видеопоток был закодирован в виде модуля  
из одного или множества кадров и записан на носитель записи, и в котором второй флаг  
был записан в модуле кодирования, причем второй флаг представляет, действительно ли  
40 текущий угол можно переключить в начале модуля кодирования.

Как описано выше, в соответствии с настоящим изобретением, поскольку на  
носителе записи были записаны видеопоток, первый флаг, который описан в каждом  
45 воспроизводимом блоке и который представляет, действительно ли каждый  
воспроизводимый блок может быть воспроизведен с множеством углов, количество углов  
в каждом воспроизводимом блоке, и информация о положении, которая представляет  
положение каждого угла в видеопотоке, причем видеопоток, который составляет угол,  
50 может считываться в соответствии с информацией о положении, в котором видеопоток



был закодирован в виде модуля из одного или множества кадров и записан на носитель записи, и в котором второй флаг был записан в модуле кодирования, причем второй флаг представляет, действительно ли текущий угол можно переключить в начале модуля кодирования. Таким образом, при воспроизведении видеопотока каждого воспроизводимого блока, текущий угол обзора можно переключать на другой угол.

### Краткое описание чертежей

На фиг. 1 показана схема, изображающая механизм, предназначенный для определения порядка воспроизведения файла AV потока;

на фиг. 2 показана схема УЯМ (UML унифицированный язык моделирования), изображающая соотношение AV потока клипа, информации клипа, клипа, элемента воспроизведения и списка воспроизведения;

на фиг. 3 показана схема, описывающая способ ссылки на тот же клип из множества списков воспроизведения;

на фиг. 4 показана схема, описывающая структуру управления файлами, записанными на носитель записи;

на фиг. 5 показана схема, изображающая синтаксис, который описывает пример структуры файла "info.bdav";

на фиг. 6 показана схема, изображающая синтаксис, который описывает пример структуры блока UIAppInfoBDAV();

на фиг. 7 показана схема, описывающая синтаксис, который описывает пример структуры блока TableOfPlayLists();

на фиг. 8 показана схема, изображающая синтаксис, который описывает пример структуры файлов "#####.rpls" и "#####.vpls";

на фиг. 9 показана схема, изображающая синтаксис, который описывает пример структуры блока UIAppInfoPlayList();

на фиг. 10 показана схема, изображающая синтаксис, который описывает пример структуры блока PlayList();

на фиг. 11 показана схема, изображающая синтаксис, который описывает пример структуры блока PlayItem();

на фиг. 12 показана схема, описывающая мостовой клип;

на фиг. 13 показана схема, изображающая синтаксис, который описывает пример структуры блока PlayListMark();

на фиг. 14 показана схема, изображающая синтаксис, который описывает пример структуры файла "%%%%%%.dpi";

на фиг. 15 показана схема, изображающая пример структуры плана, используемого в качестве системы представления изображения в соответствии с вариантом выполнения настоящего изобретения;

на фиг. 16 показана схема, изображающая примеры разрешающей способности и отображаемых цветов плана подвижного изображения, плана подзаголовка, и плана графического изображения;

на фиг. 17 показана блок-схема, изображающая пример структуры, в которой комбинируется план подвижного изображения, план подзаголовка, и план графического изображения;

на фиг. 18 показана схема, изображающая пример входных и выходных данных палитры;

на фиг. 19 показана схема, изображающая пример таблицы палитры, содержащейся в палитре;

на фиг. 20 показана схема, изображающая пример экрана меню, отображаемого в плане графического изображения;

на фиг. 21 показана схема, изображающая пример внутренней структуры сценария в соответствии с первоначально определенным языком описания сценария;

на фиг. 22 показана схема, описывающая категории структуры сценария;

на фиг. 23А-С показаны схемы, представляющие категории структур сценариев;

на фиг. 24 показана схема, описывающая заголовки и разделы;

на фиг. 25 показана схема, описывающая модель виртуального проигрывателя ДГ (ВД - диска типа голубой луч);

на фиг. 26А и на фиг. 26В показаны схемы последовательности выполнения операций, описывающие работу виртуального проигрывателя 30 ДГ в соответствии с командами, описанными в сценарии;

на фиг. 27А и на фиг. 27В показаны схемы, последовательности выполнения операций при воспроизведении в соответствии со списками воспроизведения;

на фиг. 28А-Н показаны схемы, изображающие примеры команд, используемых в сценарии;

на фиг. 29 показана схема, изображающая пример структуры управления файлами;

на фиг. 30 показана схема, изображающая синтаксис, который описывает пример структуры файла "scenario.hdmv";

на фиг. 31 показана схема, изображающая синтаксис, который описывает пример структуры данных блока Autoplay();

на фиг. 32 показана схема, изображающая синтаксис, который описывает пример

структуры блока Scenario());

на фиг. 33 показана схема, изображающая синтаксис, который описывает пример структуры данных файла "entrylist.data";

5

на фиг. 34 показана схема, изображающая синтаксис, который описывает пример структуры блока AppInfo());

10

на фиг. 35 показана схема, изображающая синтаксис, который описывает пример структуры блока ScenarioEntry());

на фиг. 36 показана схема, изображающая синтаксис, который описывает пример структуры файла "xxxx.mpls";

15

на фиг. 37 показана схема, изображающая синтаксис, который описывает пример структуры блока PLCpntrolInfo());

на фиг. 38 показана схема, которая описывает поле PL\_Playback\_type;

на фиг. 39 показана схема, которая описывает поле PL\_random\_access\_mode;

20

на фиг. 40 показана схема, изображающая синтаксис, который описывает пример структуры блока PlayList());

на фиг. 41 показана схема, изображающая синтаксис, который описывает пример структуры блока PlayItem());

25

на фиг. 42 показана схема, описывающая поле PI\_random\_access\_mode;

на фиг. 43 показана схема, описывающая поле still\_mode;

на фиг. 44 показана схема, описывающая поле is\_seamless\_angle\_change;

30

на фиг. 45 показана схема, изображающая синтаксис, который описывает пример структуры блока SubPlayItem());

на фиг. 46 показана схема, описывающая поле is\_repeat\_flag;

35

на фиг. 47 показана схема, описывающая синхронное воспроизведение для элемента воспроизведения подпотока в соответствии с основным элементом воспроизведения;

40

на фиг. 48 показана схема, изображающая синтаксис, который описывает пример структуры файла "zzzzz.clpi";

на фиг. 49 показана схема, изображающая синтаксис, который описывает пример структуры блока ClipInfo());

45

на фиг. 50 показана схема, описывающая поле application\_type;

на фиг. 51 показана схема, изображающая синтаксис, который описывает пример структуры блока SequenceInfo());

50

на фиг. 52 показана схема, изображающая синтаксис, который описывает пример структуры блока ProgramInfo());

на фиг. 53 показана схема, изображающая синтаксис, который описывает пример структуры блока StreamCodingInfo();

на фиг. 54 показана схема, изображающая синтаксис, который описывает пример структуры блока CPI();

на фиг. 55 показана схема, описывающая поле CPI\_type;

на фиг. 56 показана схема, изображающая синтаксис, который описывает пример структуры блока EP\_map\_for\_HDMV();

на фиг. 57 показана схема, описывающая протяженность;

на фиг. 58 показана схема, изображающая, пример фрагментированной записи на диске АВ потока клипа;

на фиг. 59А и на фиг. 59В показаны схемы, представляющие функцию множества углов;

на фиг. 60А и на фиг. 60В показаны схемы, представляющие множество точек переключения углов, описанных в одном блоке чередования;

на фиг. 61 показана схема, описывающая флаг is\_angle\_change\_point;

на фиг. 62А и на фиг. 62В показаны схемы, изображающие пример небезстыкового блока множества углов;

на фиг. 63А и на фиг. 63В показаны схемы, представляющие, что при включении угла в небезстыковом блоке множества углов, имеет место разрыв;

на фиг. 64А и на фиг. 64В показаны схемы, представляющие первый способ воспроизведения для небезстыкового блока множества углов;

на фиг. 65А и на фиг. 65В показаны схемы, представляющие второй способ воспроизведения для небезстыкового блока множества углов; и

на фиг. 66А-С показаны функциональные блок-схемы, представляющие пример структуры декодера 100 устройства воспроизведения в соответствии с вариантом выполнения настоящего изобретения.

#### **Подробное описание изобретения**

Далее будет описан вариант выполнения настоящего изобретения. В соответствии с вариантом выполнения настоящего изобретения, который основан на стандарте диска типа "голубой луч" (формат перезаписываемого диска типа "голубой луч", версия 1.0), который является стандартом, определяющим порядок записи и воспроизведения данных, функции, необходимые для диска, предназначенного только для воспроизведения, обеспечивающие выполнение интерактивной функции и функции множества углов.

Прежде всего, для простоты понимания настоящего изобретения, будет описана

структура управления, в соответствии со спецификацией "формата перезаписываемого диска типа "голубой луч", версия 1.0 часть 3 спецификация аудио- и видеоданных содержания, а именно, АВ (аудио/видео) данных, заранее записанных на диск типа "голубой луч". В следующем описании, структура управления называется форматом BDAV – аудио/видео данных для диска типа "голубой луч".

Поток битов, который был закодирован в соответствии с системой кодирования типа видео-MPEG (Экспертная группа по вопросам движущегося изображения) или аудио-MPEG и мультиплексирован в соответствии с системой MPEG-2, называется АВ потоком клипа (или просто АВ потоком). АВ поток клипа записан в виде файла на диск с использованием файловой системы, определенной в спецификации "формата перезаписываемого диска типа "голубой луч", Версия 1.0 часть 2" для диска типа "голубой луч". Этот файл называется файлом АВ потока клипа (или просто АВ потоком).

Файл АВ потока клипа представляет собой модуль управления для файловой системы. При этом нельзя сказать, что файл АВ потока клипа является модулем управления, который легко понятен для пользователя. С точки зрения удобства пользователя, необходимо записывать информацию, необходимую для объединения видеосодержания, которое было разделено на множество файлов АВ потока клипа, и воспроизводить объединенное видеосодержание, информацию, необходимую для воспроизведения только части файла АВ потока клипа, информацию, необходимую для ровного выполнения специального воспроизведения и воспроизведения поиска и т.д. в виде базы данных. Спецификация "формат перезаписываемого диска типа "голубой луч", Версия. 1.0 часть 3", представляющая собой стандарт для диска типа "голубой луч", определяет такую базу данных.

На фиг. 1 схематично показан механизм, предназначенный для обозначения части всего файла АВ потока, упорядочивающий только необходимые его части и воспроизводящий упорядоченные части. На фиг. 1, список воспроизведения (PlayList) обеспечивает обозначение части или всего файла АВ потока и воспроизведение только требуемых его частей. Когда пользователь воспроизводит содержание, он или она выбирает его в модуле списка воспроизведения. Список воспроизведения представляет собой один видео/аудио модуль, в котором пользователь неявным образом ожидает, что содержание будет воспроизводиться последовательно.

Простейшая структура списка воспроизведения состоит из одного файла АВ потока от начала записи содержания, до ее остановки. Если только файл АВ потока не будет отредактирован, он становится одним списком воспроизведения

Список воспроизведения состоит из информации, которая представляет файл АВ

потока, предназначенный для воспроизведения, и устанавливает наборы точек начала воспроизведения и точек остановки воспроизведения, которые обозначают положения начала воспроизведения и положения остановки воспроизведения файла АВ потока. Пара данных точки начала воспроизведения и данных точки остановки воспроизведения называется элементом воспроизведения (PlayItem). Список воспроизведения состоит из набора элементов воспроизведения. При воспроизведении элемента воспроизведения воспроизводится часть файла АВ потока, на которую делается ссылка из элемента воспроизведения.

Как описано выше, АВ поток клипа представляет собой битовый поток, в котором видеоданные и аудиоданные были мультиплексированы в формате MPEG2 ТП (TS транспортный поток). Информацию об АВ потоке клипа записывают в виде информации клипа в файл.

Набор из файла АВ потока клипа и файла информации клипа, который содержит соответствующую информацию клипа, обрабатывают как один объект и обозначают как клип. Клип представляет собой один объект, который состоит из АВ потока клипа и информации клипа.

Файл обычно обрабатывают как последовательность байтов. Содержание файла АВ потока клипа расширяют на временной оси. Точка входа в клип обычно обозначается по времени. Когда задан штамп времени точки входа для определенного клипа, файл информации клипа можно использовать для поиска информации адреса, с которого данные считывают в файле АВ потока клипа.

Все списки воспроизведения и клипы, записанные на диске, управляются с помощью информации объема.

На фиг. 2 показана схема УЯМ (унифицированного языка моделирования), которая представляет взаимоотношение предыдущего АВ потока клипа, информации клипа (атрибуты потока), клипов, элементов воспроизведения и списка воспроизведения. Один список воспроизведения скоррелирован с одним или множеством элементов воспроизведения. Один элемент воспроизведения коррелирован с одним клипом. Один клип может быть скоррелирован с множеством элементов воспроизведения, начальные точки которых и/или конечные точки которых являются различными. Ссылка на один файл АВ потока клипа делается из одного клипа. Ссылка на один файл информации клипа делается из одного клипа. Один файл АВ потока клипа и один файл информации клипа скоррелированы в соотношении один к одному. При определении такой структуры порядок воспроизведения может быть неструктивно обозначен путем воспроизведения только одной части, без изменения файла АВ потока клипа.

Как показано на фиг. 3, ссылка на один и тот же клип может быть сделана из множества списков воспроизведения. В примере, показанном на фиг. 3, ссылка на клип 1 делается из двух списков 2 и 3 воспроизведения. На фиг. 3 горизонтальное направление клипа 1 представляет линию времени. Из списка 2 воспроизведения делаются ссылки на области от а до f клипа 1, которые включают области b и c коммерческого сообщения (КС) и сцену е. Из списка 3 воспроизведения делается ссылка на область от d до g клипа 1, которая включает сцену е. Когда обозначается список 2 воспроизведения, может воспроизводиться область от а до f клипа 1. При обозначении списка 3 воспроизведения обеспечивается воспроизведение области от d до g клипа 1.

Далее, со ссылкой на фиг. 4 будет описана структура управления для файлов, записанных на носитель записи, описанная в формате перезаписываемого диска типа "голубой луч", версия 1.0 часть 3". Управление файлами выполняется иерархически в структуре директории. Одну директорию (корневую директорию в примере, показанном на фиг. 4) создают на носителе записи. В этой директории управление файлами обеспечивается с помощью одной системы записи и воспроизведения.

В корневой директории размещена директория BDAV. Как показано на фиг. 4, может быть размещено множество директорий, таких, как директории BDAV, BDAV1, BDAV2.... BDAVn. В следующем описании, множество директорий BDAV, BDAV1, BDAV2..... и BDAVn представлены директорией DAV. Будет описана только представительная директория BDAV.

В директории BDAV размещены следующие шесть типов файлов.

- (1) info.bdav
- (2) menu.tidx, mark.tidx
- (3) menu.tdt1, menu.tdt2, mark.tdt1, mark.tdt2
- (4) #####.rpls, #####.vpls
- (5) %%%%.clpi
- (6) \*\*\*\*\*.m2ts

В файлах #####.rpls" и "#####.vpls", в соответствующих категориях (4), "#####" представляет любое число. В файле "%%%.clpi" из категорий (5), "%%%.clpi" представляет любое число. В файле "\*\*\*\*\*.m2ts" из категорий (6), "\*\*\*\*\*" представляет любое число, в которых файл "\*\*\*\*\*.m2ts" соответствует файлу "%%%.clpi" в отношении один к одному. Число "\*\*\*\*\*" может быть тем же, что и число "%%%.clpi".

Файл "info.bdav" из категории (1) представляет собой файл, который имеет информацию обо всей директории BDAV. Файлы "menu.tidx" и "mark.tidx" из категории (2) представляют собой файлы, которые содержат информацию об уменьшенных

изображениях. Файлы "menu.tdt1", "menu.tdt2", "mark.tdt1" и "mark.tdt2" из категории (3) представляют собой файлы, которые содержат уменьшенное изображение. Расширение "tdt1" и "tdt2" этих файлов представляют, были или нет данные этих уменьшенных изображений в этих файлах зашифрованы.

Файлы "#####.rpls" и "#####.vpls" из категории (4) представляют собой файлы, которые содержат информацию о списках воспроизведения. Файлы "#####.rpls" и "#####.vpls" расположены в директории PLAYLIST, которая расположена в директории BDAV.

Файл "%%%%%%.clpi" из категории (5) представляет собой файл, который содержит информацию клипа. Файл "%%%%%%.CLP" расположен в директории CLIPINF, который расположен в директории BDAV. Файл "\*\*\*\*\*.m2ts" из категории (6) представляет собой файл АВ потока клипа, который содержит АВ поток клипа. Файл АВ потока клипа коррелирован с одним файлом информации клипа "%%%%%%.clpi" с файлом, название которого представляет собой номер "\*\*\*\*\*". Файл "\*\*\*\*\*.m2ts" расположен в директории STREAM, которая расположена в директории BDAV.

Каждый файл будет подробно описан ниже. Файл "info.bda" из категории (1) представляет собой только один файл, расположенный в директории BDAV. На фиг. 5 показан синтаксис, который описывает пример структуры файла "info.bdav". Синтаксис представлен описательным методом языка C, который используется как язык описания программы для компьютерных устройств. Это относится к чертежам, на которых представлены другие синтаксисы.

На фиг. 5 файл "info.bdav" разделен на блоки, соответствующие функциям. Поле `type_indicator` описывает строку знаков "BDAV", которая описывает, что этот файл представляет собой "info.bdav". Поле `version_number` представляет версию файла "info.bdav". Блок `UIAppInfoBDAV()` описывает информацию об информации, размещенной в директории DBAV. Блок `TableOfPlayList()` описывает информацию о компоновке списка воспроизведения. Блок `MakersPrivateData()` описывает уникальную информацию изготовителя устройства записи и воспроизведения.

Адреса, которые представляют начальные точки отдельных блоков, записаны в начале файла "info.bdav". Например, поле `TableOfPlayLists_Start_address` представляет начальное положение блока "TableOfPlayLists()" с количеством относительных байтов в файле.

На фиг. 6 показан синтаксис, который описывает пример структуры блока `UIAppInfoBDAV()`. Длина поля представляет длину непосредственно после длины поля до конца блока `UIAppInfoBDAV()` в байтах. Поле `BDAV_character_set` представляет набор



знаков последовательности знаков, описанной в поле BDAV\_name блока UIAppInfoBDAV(). В качестве набора знаков можно выбрать код ASCII (американский стандартный код для обмена информацией), Unicode (стандарт ISO 10646 кодирования символов всех национальных алфавитов.) или подобный код.

Флаг BDAV\_protect\_flag описывает, получено или нет от пользователя безусловное разрешение на просмотр содержания, размещенного в директории BDAV. Когда флаг установлен в "1", и пользователь ввел правильный PIN (личный идентификационный номер), он или она получает разрешение на просмотр содержания, размещенного в директории BDAV. В отличие от этого, когда флаг BDAV\_protect\_flag был установлен в "0", даже если пользователь не вводит свой PIN, ему или ей разрешается просматривать содержание, размещенное в директории BDAV.

Личный идентификационный номер PIN описан в поле PIN. Личный идентификационный номер PIN представляет собой, например, число из четырех цифр, каждая цифра в диапазоне от 0 до 9. Личный идентификационный номер PIN представляет собой личный идентификационный номер, который требуется, когда выполняется разрешение на управление воспроизведением. Цифры личного идентификационного номера PIN закодированы в соответствии, например, со стандартом 646 ISO (Международная организация по стандартизации)/IEC (Международная электротехническая комиссия).

С помощью вышеуказанной информации, описанной в блоке UIAppInfoBDAV(), определяется ограничение по воспроизведению для директории BDAV. Как будет описано ниже, ограничения по воспроизведению для каждого списка воспроизведения описаны с помощью флага playback\_control\_flag, который определен в блоке UIAppInfoPlayList(), описанном в файлах "#####.rpls" и "#####.vpls".

В примере для возобновления воспроизведения содержания, расположенного в директории BDAV, можно использовать функцию возобновления. Функция возобновления позволяет назначать приоритеты для списка воспроизведения воспроизводимого содержания. Предполагается, что функция возобновления используется, когда пользователь желает возобновить воспроизведение содержания от последнего положения остановки.

На фиг. 6 флаг resume\_valid\_flag описывает, является ли функция возобновления разрешенной/не разрешенной. Когда значение флага установлено в "0", функция возобновления является не разрешенной. Когда значение флага установлено в "1", функция возобновления является разрешенной. В этой точке список воспроизведения, обозначаемый полем resume\_PlayList\_file\_name, обрабатывается как список

воспроизведения, для воспроизведения с приоритетом.

Поле `ref_to_menu_thumbnail_index` представляет собой область, которая описывает номер уменьшенного изображения, идентифицирующий уменьшенное изображение, которое характеризует тип директории BDAV. В стандарте диска типа "голубой луч" неподвижное изображение, которое характеризует тип директории BDAV, обозначается как уменьшенное изображение меню. Уменьшенное изображение, которое имеет индекс `thumbnail_index`, описанный в поле `ref_to_menu_thumbnail_index`, представляет собой уменьшенное изображение меню директории BDAV.

Поле `BDAV_name_length` представляет длину в байтах названия директории BDAV, описанного в поле `BDAV_name`. Количество байтов, представленных в поле `BDAV_name_length`, является действительным для строки знаков поля `BDAV_name`, которая представляет название директории BDAV. Остальная часть последовательности байтов после разрешенной строки знаков, представленная полем `BDAV_name_length`, может иметь любое значение.

На фиг. 7 показан синтаксис, который описывает пример структуры блока `TableOfPlayLists()`. Поле `number_of_PlayLists` представляет собой количество списков воспроизведения, расположенных в директории BDAV. Количество списков воспроизведения обозначается как переменная цикла. После поля `number_of_PlayLists` следует цикл выражения "for". Выражение "for" описывает списки воспроизведения, представленные полем `PlayList_file_name`. Списки воспроизведения отображаются на экране отображения таблицы списка воспроизведения или подобном. Список воспроизведения обозначается названием файла, например, "#####.rpls" или "#####.vpls" в поле `PlayList_file_name`.

Как описано выше, файлы "#####.rpls" и "#####.vpls" расположены в директории PLAYLIST. Эти файлы соответствуют отдельным спискам воспроизведения в соотношении один к одному.

На фиг. 8 показан синтаксис, который описывает пример структуры файлов "#####.rpls" и "#####.VPLS". На фиг. 8 каждый из файлов "#####.rpls" и "#####.vpls" имеет блоки, соответствующие функциональной информации. Поле `type_indicator` описывает строку знаков, которая представляет файл. Поле `version_number` представляет версию файла.

Блок `UIAppInfoPlayList()` описывает информацию атрибута списка воспроизведения. Блок `PlayList()` описывает информацию о пунктах воспроизведения, которые составляют список воспроизведения. Блок `PlayListMark()` описывает информацию о метке, добавленной к списку воспроизведения. Блок `MakersPrivateData()` описывает

уникальную информацию производителя устройства, который записал файл списка воспроизведения. Поля `Playlist_start_address`, `PlaylistMark_start_address` и `MakersPrivateData_start_address` расположены в начале каждого из файлов "#####.rpls" и "#####.vpls". Эти поля описывают начальные адреса соответствующих блоков информации адреса из 32 битов.

Поскольку адрес начала каждого блока описан в начале каждого из файлов "#####.rpls" и "#####.vpls", данные `padding_word` любой длины могут быть расположены перед каждым блоком и/или после каждого блока. Однако, начальное положение блока `UIAppInfoPlaylist()`, который представляет собой первый блок каждого из файлов "#####.rpls" и "#####.vpls", является фиксированным на 320-ом байте от начала каждого из этих файлов.

На фиг. 9 показан синтаксис, который описывает пример структуры блока `UIAppInfoPlaylist()`. Блок `UIAppInfoPlaylist()` описывает различные типы информации атрибута о списке воспроизведения. Информация атрибута не используется непосредственно для воспроизведения списка воспроизведения. Поле `Playlist_character` описывает набор знаков информации строки знаков о списке воспроизведения.

Флаг `playback_control_flag` описывает, являются или нет отображение информации и воспроизведение списка воспроизведения ограниченными в соответствии с личным идентификационным номером PIN. Когда значение флага `playback_control_flag` равно, например, "1", до тех пор, пока пользователь не введет правильный личный идентификационный номер PIN, информация, такая, как уменьшенное изображение списка воспроизведения не может отображаться, и список воспроизведения не может воспроизводиться. Флаг `write_protect_flag` представляет собой флаг запрета стирания. Для структуры интерфейса пользователя необходимо, чтобы пользователь не мог легко стереть список воспроизведения, когда значение флага `write_protect_flag` равно "1". Флаг `is_played_flag` описывает, что список воспроизведения был воспроизведен. Флаг `is_edited_flag` описывает, что список воспроизведения был отредактирован.

Поле `time_zone` представляет временную зону, в которой был записан список воспроизведения. Поле `record_time_and_date` представляет дату и время записи списка воспроизведения. Поле `Playlist_duration` представляет длительность воспроизведения списка воспроизведения.

Поля `maker_ID` и `maker_model_code` описывают информацию, которая идентифицирует изготовителя и модель устройства записи для последнего обновления списка воспроизведения. Поля `maker_ID` и `maker_model_code` представляют собой, например, числа. Поле `channel_number` представляет номер канала записанного АВ потока

клипа. Поле `channel_name` представляет название канала. Поле `channel_name_length` представляет длину названия канала, описанного в поле `channel_name`. В поле `channel_name`, строка знаков, имеющая длину, описанную в поле `channel_name_length`, является разрешенной. Поле `PlayList_name` представляет название списка воспроизведения, значение эффективной длины которого описано в поле `PlayList_name_length`. Поле `PlayList_detail` описывает подробную информацию списка воспроизведения, имеющую значение эффективной длины, описанное в поле `PlayList_detail_length`.

На фиг. 10 показан синтаксис, который описывает пример структуры блока `PlayList()`. Поле `length` (длина) описывает длину в байтах непосредственно после поля `length` до конца блока `PlayList()`. Поле `PL_CPI_type` описывает тип CPI (информации характеристической точки) списка воспроизведения. Поле `number_of_PlayItems` описывает количество элементов воспроизведения, которые составляют список воспроизведения. Поле `number_of_SubPlayItems` описывает количество элементов воспроизведения для звуковых данных, добавленных к списку воспроизведения после записи (подпоток элементов воспроизведения). Вкратце, список воспроизведения может иметь подпоток элемента воспроизведения, когда список воспроизведения удовлетворяет заданному условию.

Блок `PlayItem()` описывает информацию элемента воспроизведения. Блок `SubPlayItem()` описывает информацию подпотока элемента воспроизведения.

На фиг. 11 показан синтаксис, который описывает пример структуры блока `PlayItem()`. Поле `Clip_Information_file_name` описывает строку знаков названия файла информации клипа (то есть, файла, имеющего расширение `clpi`), который находится в соотношении один к одному с клипом, на который ссылается элемент воспроизведения. Файл информации клипа представляет собой файл, имеющий расширение `"clpi"`.

Поле `Clip_codec_identifier` описывает систему кодирования клипа, на которую ссылается элемент воспроизведения. В примере, в поле `Clip_codec_Identifier` зафиксировано значение `"M2TS"`. Поле `connection_condition` описывает информацию о соединении этого элемента воспроизведения со следующим элементом воспроизведения. Поле `connection_condition` описывает, могут или нет элементы воспроизводиться без стыков.

Поле `ref_to_STC_id` обозначает последовательность `STC_sequence` клипа, на которую ссылается элемент воспроизведения. Последовательность `STC_sequence` представляет собой уникальную структуру стандарта диска типа "голубой луч". Эта структура представляет диапазон, в котором ТЧП (PCR - опорная тактовая частота

программы), которая представляет собой опорную частоту MPEG2 TS (ТП Транспортного Потока), является непрерывной на временной оси. Число STC\_id, которое является уникальным в клипе, назначают последовательности STC\_sequence. В последовательности STC\_sequence, поскольку может быть определена непрерывная временная ось, время начала и время окончания элемента воспроизведения может быть уникально обозначено. Начальная точка и конечная точка каждого элемента воспроизведения должны существовать в одной последовательности STC\_sequence. Поле ref\_to\_STC\_id описывает последовательность STC\_sequence с числом STC\_id.

Поля IN\_time и OUT\_time описывают временные штампы pts (presentation\_time\_stamp) начальной точки и конечной точки элемента воспроизведения в последовательности STC\_sequence, соответственно.

Блок BridgeSequenceInfo() описывает информацию о мостовом клипе (Bridge\_Clip). Как показано на фиг. 12, мостовой клип представляет собой поток битов, который создается при выполнении функции воспроизведения без стыков элементов воспроизведения. Благодаря воспроизведению мостового клипа вместо исходного потока битов два элемента воспроизведения могут воспроизводиться без стыков на границе предыдущего элемента воспроизведения и текущего элемента воспроизведения. Поскольку функция мостового клипа не относится к настоящему изобретению, ее описание здесь не приводится.

На фиг. 13 показан синтаксис, который описывает пример структуры блока PlaylistMark(). Блок PlaylistMark() имеет структуру данных, которая описывает информацию о метке. Метка представляет собой структуру, которая описывает время списка воспроизведения. С помощью метки выполняется функция установки точки поиска в списке воспроизведения, функция разделения списка воспроизведения на разделы, и т. д. Время начала отображения и останковки отображения изображения на плане графического изображения (которая будет описана ниже), может быть обозначено с помощью меток.

Поле length описывает длину в байтах непосредственно после поля length до конца блока Playlistmark(). Поле number\_of\_PlayList\_marks описывает количество меток в списке воспроизведения. Один цикл выражения "for" представляет информацию об одной метке. Флаг mark\_invalid\_flag описывает, является ли метка разрешенной или нет. Когда значение флага mark\_invalid\_flag равно "0", он описывает, что метка является разрешенной. Когда значение флага mark\_invalid\_flag установлено в "1", он описывает, что, хотя информация метки присутствует в базе данных, эта метка является не разрешенной меткой, которая является прозрачной для пользователя.

Поле mark\_type описывает тип метки. Существуют метка, которая представляет

положение изображения в виде уменьшенного изображения (представление изображения)  
списка воспроизведения, метка продолжения, которая представляет положение, от  
которого продолжается воспроизведение, метка раздела, которая представляет точку  
5 поиска, метка пропуска, которая представляет область для пропуска и воспроизведения,  
метка, которая представляет синхронизацию начала считывания графического  
изображения, метка, которая представляет время начала отображения графического  
10 изображения, метка, которая представляет время остановки отображения графического  
изображения, и т. д.

Поле `mark_name_length` представляет длину данных поля `mark_name` (которое  
будет описана ниже). Поле `maker_ID` описывает изготовителя устройства записи, который  
15 создал метку. Поле `maker_ID` используется для идентификации уникальной для  
изготовителя метки. Поле `ref_to_PlayItem_id` описывает, какой элемент воспроизведения  
имеет время, обозначенное меткой. Поле `mark_time_stamp` представляет время,  
20 обозначенное меткой.

Поле `entry_ES_PID` описывает, в какой элементарный поток была добавлена метка  
(а именно, была ли метка добавлена к потоку, в котором были закодированы данные  
25 изображения и/или данные звука). Поле `ref_to_menu_thumbnail_index` и поле  
`ref_to_mark_thumbnail_index` описывают уменьшенные изображения, которые визуально  
представляют метки. Уменьшенное изображение представляет собой, например,  
неподвижное изображение, которое было выделено в момент времени, обозначенный  
30 меткой.

Длительность поля используется, когда метка имеет длину на временной оси.  
Когда используется метка пропуска, длительность поля описывает длительность  
35 пропуска.

Поле `makers_information` представляет собой область, которая описывает  
информацию, уникальную для изготовителя. Поле `mark_name` представляет собой  
40 область, которая описывает название, назначенное метке. Размер метки описан в  
указанном выше поле `mark_name_length`.

На фиг. 14 показан синтаксис, который описывает пример структуры файла  
"%%%%.clpi". Как описано выше, файл "%%%%.clpi" расположен в директории  
45 CLIPINF. Файл "%%%%.clpi" создан для каждого файла АВ потока (файл "\*\*\*\*\*.m2ts").  
Файл "%%%%.clpi" содержит блоки, соответствующие функциональной информации.  
Поле `type_indicator` описывает строку знаков, которая представляет файл. Поле  
50 `version_number` описывает версию файла.

Блок `ClipInfo()` описывает информацию о клипе. Блок `SequenceInfo()` описывает

информацию о точке, содержащей разрыв ТЧП, который представляет временную ссылку транспортного потока системы MPEG2. Блок ProgramInfo() описывает информацию о программе системы MPEG2. Блок CPI() описывает информацию об информации характеристической точки CPI, которая представляет собой характеристический участок АВ потока. Блок ClipMark() описывает информацию метки, которая представляет точку индекса поиска, добавленную к клипу, и точки коммерческого начала и/или окончания. Блок MakersPrivateData() описывает информацию, уникальную для изготовителя устройства записи.

Информация адреса, которая представляет начало каждого блока в файле "%%%%.clpi", описывает поля SequenceInfo\_start\_address, ProgramInfo\_start\_address, CPI\_start\_address, ClipMark\_start\_address, и MakersPrivateData\_start\_address. Поскольку файл "%%%%.clpi" клипа не относится к настоящему изобретению, его описание здесь не приведено.

Поскольку формат BDAV имеет описанную выше структуру данных, с помощью списка воспроизведения, состоящего из элементов воспроизведения, которые описывают наборы начальных точек и конечных точек участков, предназначенных для воспроизведения в АВ потоке клипа, для пользователя обеспечивается возможность управления содержанием, записанным на диск, в блоке воспроизведения.

Далее будет описан вариант выполнения настоящего изобретения. В соответствии с настоящим изобретением, указанный выше формат BDAV расширен на формат диска, предназначенного только для воспроизведения. Вначале будет описана структура плана, в котором формируется экран меню содержания диска. Затем добавлена структура сценария, которая позволяет производителю содержания обозначать порядок воспроизведения списка воспроизведения. Для структуры сценария описаны данные, необходимые для выполнения таких функций, как воспроизведение неподвижного изображения (пауза), воспроизведение в случайном порядке, воспроизведение из множества углов обзора и т. д., которые характерны для диска, предназначенного только для воспроизведения, и способ записи таких данных.

В соответствии с вариантом выполнения настоящего изобретения используется структура плана, показанная на фиг. 15. План 10 подвижного изображения отображается на самой задней стороне (на дне). План 10 подвижного изображения относится к изображению (в основном, к данным подвижного изображения), определяемому в соответствии со списком воспроизведения. План 11 подзаголовка отображается над планом 10 подвижного изображения. План 11 подзаголовка относится к данным подзаголовка, отображаемым при воспроизведении подвижного изображения. План 12

графического изображения отображается спереди. План 12 графического изображения относится к знаковым данным экрана меню и графических изображений, таким, как данные растрового изображения кнопок. Один экран отображения состоит из трех таких планов.

Различия между вариантами выполнения в соответствии с настоящим изобретением и обычным видеодиском DVD состоит в том, что изображения подзаголовков, экрана меню, кнопок и т. д. выделены в план 11 подзаголовка и план 12 графических изображений, что обеспечивает независимое управление подзаголовками и кнопками. В обычном видеодиске DVD, графическими изображениями такими, как экран меню и кнопки, а также изображение подзаголовка управляют с помощью одного механизма. Их отображают в одном плане. Количество одновременно отображаемых растровых изображений ограничено единицей. При этом в видеодиске DVD множество растровых изображений нельзя отображать одновременно. В отличие от этого, в соответствии с настоящим изобретением, поскольку план 11 подзаголовка и план 12 графических изображений расположены, соответственно, независимо для подзаголовков и графических изображений, вышеуказанная проблема обычного DVD может быть решена.

Можно считать, что план 11 подзаголовков и план 12 графических изображений представляют собой часть расширения обычного "формата перепереписываемого диска типа "голубой луч", версия 1.0, часть 3".

План 10 подвижного изображения, план 11 подзаголовка и план 12 графического изображения можно отображать независимо. План 10 подвижного изображения, план 11 подзаголовка и план 12 графического изображения имеют разрешающую способность и цвета отображения, показанные на фиг. 16. План 10 подвижного изображения имеет разрешающую способность 1920 пикселей x 1080 строк, длину данных 16 битов на пиксель, систему цветности YCbCr (4 : 2 : 2), где Y представляет сигнал яркости, и Cb и Cr представляют цветоразностные сигналы. YCbCr (4 : 2 : 2) представляет систему цветности, содержащую сигнал Y яркости из восьми битов на пиксель и цветоразностные сигналы Cb и Cr по восемь битов каждый. Данные одного цвета состоят из двух горизонтальных пикселей цветоразностных сигналов Cb и Cr.

План 12 графического изображения имеет разрешающую способность 1920 пикселей x 1080 строк, глубину выборки восемь битов на пиксель, и систему цветности с адресами восьмибитовой структуры цвета, с использованием палитры из 256 цветов. План 11 подзаголовка имеет разрешающую способность 1920 пикселей x 1080 строк, глубину выборки восемь битов на пиксель и систему цветности, имеющую адреса восьмибитовой структуры цвета, с использованием палитры из 256 цветов.



План 12 графических изображений и план 11 подзаголовка могут быть альфа-смешанными на 256 уровнях. Когда план 12 графических изображений и план 11 подзаголовка комбинируют с другим планом, степень прозрачности может быть установлена на 256 уровнях. Степень прозрачности может быть установлена для каждого пикселя. В следующем описании степень альфа-прозрачности представлена в диапазоне ( $0 \leq \alpha \leq 1$ ), где степень  $\alpha = 0$  прозрачности представляет идеально прозрачный; степень  $\alpha = 1$  прозрачности представляет идеально не прозрачный.

План 11 подзаголовка относится к данным изображения формата ПСГ (PNG переносимая сетевая графика). Аналогично, план 12 графических изображений относится к данным изображения формата ПСГ. В формате ПСГ глубина выборки одного пикселя находится в диапазоне от одного бита до 16 битов. Когда глубина выборки равна восьми битам или 16 битам, альфа-канал, а именно информация о степени прозрачности (называемая альфа-данными) может быть добавлена для каждого компонента пикселя. Когда глубина выборки составляет восемь битов, степень прозрачности может быть обозначена на 256 уровнях. С помощью информации о степени прозрачности альфа-канала, выполняется альфа-смешение. При этом можно использовать изображение палитры, состоящее из до 256 цветов. Элемент (индекс) подготовленной палитры может быть представлен номером индекса.

Данные изображения, относящиеся к плану 11 подзаголовка, и плану 12 графических изображений, не ограничиваются форматом ПСГ. В качестве альтернативы можно использовать данные изображения, которые были закодированы со сжатием в соответствии с форматом JPEG (группа экспертов по машинной обработке фотографических изображений), например, данные изображения, которые были сжаты с ограничением длительности или данные растрового изображения, которые не были закодированы со сжатием.

На фиг. 17 показан пример структуры, с помощью которой три плана комбинируют со ссылкой на фиг. 15 и на фиг. 16. Данные подвижного изображения плана 10 подвижного изображения подают в схему 20 преобразования 422/444. Схема 20 преобразования 422/444 преобразует систему цветности данных подвижного изображения из YCbCr (4 : 2 : 2) в YCbCr (4 : 4 : 4) и вводит преобразованные данные в устройство 21 умножителя.

Данные изображения плана 11 подзаголовка вводят в палитру 22. Палитра 22 выводит данные изображения RGB (4 : 4 : 4). Когда определяется степень прозрачности альфа-смешения, обозначенная степень прозрачности  $\alpha_1$  ( $0 \leq \alpha_1 \leq 1$ ) поступает с выхода палитры 22.

На фиг. 18 показан пример входных/выходных данных палитры 22. Палитра 22 содержит информацию палитры в виде таблице, соответствующей, например, файлу в формате ПСГ. Номер индекса обозначен как адрес данных изображения входных данных из восьми битов, поступающих из палитры 22. В соответствии с номером индекса, на выход поступают данные RGB (4 : 4 : 4), состоящие из данных длиной восемь битов каждые. Кроме того, из палитры 22 получают данные альфа-канала, которые представляют степень прозрачности.

На фиг. 19 показан пример таблицы палитры, которую содержит палитра 22. Значения индекса 256 цветов от [0x00] до [0xFF] (где [0x] представляет шестнадцатеричное обозначение) назначают трем значениям R, G и B первичных цветов, и степень  $\alpha$  прозрачности, представленную восьмью битами. Палитра 22, делает ссылку на таблицу палитры, в соответствии с входными данными изображения в формате ПСГ, и передает на выход данные цветов R, G и B (данные RGB) и степень  $\alpha$  прозрачности размером восемь битов для каждого пикселя, в соответствии со значением индекса, определенного по данным изображения.

Данные RGB, которые поступают с выхода палитры 22, передают в схему 29 преобразования RGB/YCbCr. Схема 29 преобразования RGB/YCbCr преобразует данные RGB в сигнал Y яркости и цветоразностные сигналы Cb и Cr, каждый из которых имеет длину восемь битов (ниже их вместе называют данными YCbCr). Это происходит потому, что данные планов должны быть скомбинированы в общий формат данных. Данные объединяют в данные YCbCr, то есть, в формат данных подвижного изображения.

Данные YCbCr и данные  $\alpha 1$  степени прозрачности, которые поступают с выхода схемы 29 преобразования RGB/YCbCr, подают в устройство 23 умножителя. Устройство 23 умножителя умножает входные данные YCbCr на данные  $\alpha 1$  степени прозрачности. Результат умножения подают на один вход устройства 24 сумматора. Устройство 23 умножителя умножает каждый из сигнала Y яркости, цветоразностных сигналов Cb и Cr данных YCbCr на данные  $\alpha 1$  степени прозрачности. В устройство 21 умножителя подают дополнение  $(1 - \alpha 1)$  данных  $\alpha 1$  степени прозрачности.

Устройство 21 умножителя умножает данные подвижного изображения, которые поступают с выхода схемы 20 преобразования 422/444, на дополнение  $(1 - \alpha 1)$  данных  $\alpha 1$  степени прозрачности. Умноженный результат подают на другой вход устройства 24 сумматора. Устройство 24 сумматора складывает умноженные результаты устройства 21 умножителя и устройства 23 умножителя. В результате комбинируют план 10 подвижного изображения и план 11 подзаголовка. Результат сложения устройства 24 сумматора поступает в устройство 25 умножителя.

Так же, как для плана 11 подзаголовка, данные RGB (4 : 4 : 4) поступают как данные плана 12 графического изображения из таблицы 26А палитры, и их подают в схему 26В преобразования RGB/YCbCr. Когда система цветности данных изображения  
5 плана 12 графического изображения представляет собой RGB (4 : 4 : 4), ее преобразуют в YCbCr (4 : 4 : 4) и передают на выход схемы 27 преобразования RGB/YCbCr. Данные YCbCr, которые поступают с выхода схемы 27 преобразования RGB/YCbCr, поступают на  
10 вход устройства 28 умножителя.

Когда данные изображения плана 12 графического изображения имеют формат ПСГ, для каждого пикселя данных изображения могут быть указаны данные  $\alpha_2$  (0 ≤  $\alpha_2$  ≤ 1) степени прозрачности. Данные  $\alpha_2$  степени прозрачности подают в  
15 устройство 28 умножителя. Устройство 28 умножителя умножает каждый из сигнала Y яркости и цветоразностных сигналов Cb и Cr данных YCbCr, которые поступают со схемы 27 преобразования RGB/YCbCr, на данные  $\alpha_2$  степени прозрачности. Результат  
20 умножения устройства 28 умножителя подают на один вход схемы 29 сумматора. Дополнение (1 -  $\alpha_2$ ) данных  $\alpha_2$  степени прозрачности подают на устройство 25 умножителя.

Устройство 25 умножителя умножает сложенный результат устройства 24 сумматора на дополнение (1 -  $\alpha_2$ ) данных  $\alpha_2$  степени прозрачности. Результат умножения  
25 устройства 25 умножителя подают на другой вход устройства 27 сумматора. Устройство 27 сумматора суммирует умноженные результаты устройства 25 умножителя и устройства  
30 28 умножителя. В результате комбинируют план 12 графического изображения и комбинированный результат плана 10 подвижного изображения и плана 11 подзаголовка.

Когда степень  $\alpha$  прозрачности области, не содержащей изображение, плана 11 подзаголовка и плана 12 графического изображения назначают равной 0 ( $\alpha = 0$ ), план  
35 расположенный под этими планами 11 и 12 становится прозрачным. В результате, данные подвижного изображения плана 10 подвижного изображения можно отображать в  
40 качестве фонового изображения плана 11 подзаголовка и плана 12 графического изображения.

Структура, показанная на фиг. 17, может быть выполнена с использованием аппаратных или программных средств.  
45

При определении вышеописанных планов можно отображать экран меню и кнопки, необходимые для стандарта диска, предназначенного только для воспроизведения. Когда выбирают кнопку на экране меню, воспроизводится список воспроизведения,  
50 соответствующий этой кнопке. При этом информация о связях списков воспроизведения должна была быть записана на диск. Сценарий, который определяет связь списков

воспроизведения, будет описан ниже.

Экран, на котором для пользователя представляется предложение выполнить определенную операцию, например, экран меню, может отображаться на плане 12 графического изображения. На фиг. 20 показан пример экрана 60 меню, отображаемого на 5 плане 12 графического изображения. На экране 60 меню знаки и изображения отображаются в определенных положениях. Вместе со знаками и изображениями могут 10 быть размещены "связи" и "кнопки", которые обеспечивают для пользователя возможность выбора новых операций.

"Связь" описывает способ доступа к определенному файлу со строкой знаков или 15 данными изображения. Когда пользователь обозначает на экране строку знаков или данные изображения, например, с использованием устройства-указателя, он или она получает возможность доступа к заданному файлу в соответствии со способом доступа, определенным такой строкой знаков или данными изображения. "Кнопка" имеет три типа 20 данных изображения, которые представляют нормальное состояние, состояние выбора и нажатое состояние для "связи". Когда пользователь отмечает одно изображение кнопки, данные изображения изменяются в соответствии с состоянием, которое он или она 25 включил так, что он или она может легко распознавать текущее состояние кнопки.

Когда пользователь обозначает "связь" или "кнопку", он или она перемещает курсор по экрану с помощью указателя типа мышь и нажимает на кнопку мыши 30 (нажимает кнопку мыши несколько раз) на строке знаков или на изображении "связи", или на изображении "кнопки". Ту же операцию можно выполнять с использованием другого устройства указателя, кроме указателя типа мышь. В качестве альтернативы, пользователь может обозначать "связь" или "кнопку" с помощью устройства дистанционного 35 управления или путем нажатия на клавишу клавиатуры. При этом пользователь выбирает требуемую "связь" или "кнопку" с помощью определенной клавиши, такой, как клавиша направления, и обозначает выбранную "связь" или "кнопку" с помощью клавиши ОК или 40 подобной клавиши.

В примере, показанном на фиг. 20, заголовок 61 в виде данных изображения отображается в верхнем участке экрана 60 меню, который отображается на плане 12 графического изображения. После заголовка 61 следуют элементы 62А, 62В, 62С и 62D 45 выбора, используемые в качестве связи. Когда пользователь выбирает и обозначает один из элементов 62А, 62В, 62С и 62D выбора, нажимая на клавишу, например, пункта дистанционного управления, осуществляется доступ к файлу, связанному с обозначенным 50 элементом выбора.

Внизу экрана 60 меню отображаются кнопки 64 и 65. Вместе с кнопками 64 и 65

могут отображаться подзаголовки, и можно выбирать язык выходного звука, например, английский или японский язык. Когда активируют кнопки 64 и 65, как описано выше, обеспечивается доступ к файлам, используемым для отображения их экранов установки, и отображаются заранее определенные экраны.

В нижнем левом участке экрана 60 меню, отображается строка 63 знаков, которая описывает способ выбора элемента. Строка 63 знаков отображается на плане 12 графического изображения.

Для отображения экрана меню, как показано на фиг. 20, требуется использовать некоторый описательный язык, предназначенный для описания способа отображения экрана, информации связи и т.д. В соответствии с вариантом выполнения настоящего изобретения, в качестве используемого описательного языка, применяют исходные команды управления отображения для подзаголовков и кнопок, добавленных к системе управления, в которой команды видеонавигации DVD изменены так, что может отображаться экран меню для диска типа "голубой луч".

На экране 60 меню для вышеописанного диска типа "голубой луч", отображается таблица, содержащая, например, списки воспроизведения с данными изображения, строкой знаков, кнопками и т.д. Предполагается, что, когда обозначают определенный список воспроизведения, этот определенный список воспроизведения считывается и воспроизводится с диска.

В примере, показанном на фиг. 20, таблица списков воспроизведения отображается на экране 60 меню. В действительности, изображение и звуковое сопровождение экрана 60 меню, а также изображение и звук, которые генерируются в соответствии с выбранным элементом, на экране 60 меню, состоят из множества списков воспроизведения. Когда множество списков воспроизведения, которые составляют один пункт меню, скоррелированы, может быть определен механизм, в соответствии с которым выполняют ветвление текста. Когда выполняют ветвление текста, обеспечивается выполнение функции множества текстовых блоков, в соответствии с которой текстовые обозначения изменяются по выбору пользователя, функции автоматического воспроизведения языка, которая обеспечивает автоматическое воспроизведение нужного языка в соответствии с обозначенным языком устройства воспроизведения, и функции родителей, которая обеспечивает изменение сцен в соответствии с возрастом пользователя.

Хотя эти функции являются в особенности эффективными для записанных дисков, они не описаны в используемом в настоящее время стандарте диска типа "голубой луч", который, в основном, направлен на запись/воспроизведение телевизионных передач.

В следующем описании, структура, в соответствии с которой организовано

множество списков воспроизведения, называется сценарием. На фиг. 21 показан пример внутренней структуры сценария 70. Сценарий 70 содержит множество списков от 73А до 73М воспроизведения. Кроме того, сценарий 70 содержит два участка (экраны 80А и 80В), на которых экраны выбора раздела отображаются с планом 12 графического отображения. Экран 80А содержит графические данные 74А и список 73С воспроизведения, с помощью которого отображается экран выбора раздела. Аналогично, экран 80В содержит графические данные 74В и список 73J воспроизведения, с помощью которого отображается экран выбора раздела.

Сценарий определяет как компоновку списков воспроизведения, так и моменты синхронизации отображения, в которые их отображают на плане 12 графического изображения. Моменты синхронизации отображения списков воспроизведения на плане 12 графического изображения могут быть определены с использованием команд управления отображения, добавленных к отображаемому изображению на плане графического изображения.

В примере, показанном на фиг. 21, экран 60 меню соответствует экрану 80А сценария 70. Элемент выбора (например, элемент 62А выбора) на экране 60 меню состоит из графических изображений 74А. Когда на экране 60 меню определяют элемент 62А выбора, воспроизводится список 73D воспроизведения, который соответствует элементу выбора.

В сценарии 70, показанном на фиг. 21, при загрузке диска в устройство воспроизведения, воспроизводится список 73А воспроизведения. После воспроизведения списка 73А воспроизведения, воспроизводится список 73В воспроизведения. После воспроизведения списка 73В воспроизведения, воспроизводится список 73С воспроизведения. В результате считываются графические данные 74А, и отображается экран 80А, на котором представлено предложение для пользователя выбрать раздел текстового элемента.

После отображения экрана 80А, текстовый элемент разветвляется в соответствии с выбором пользователя. В примере, показанном на фиг. 20, когда выполняют первый выбор, отображается экран 80А. После этого последовательно воспроизводят списки 73D, 73Е и 73F воспроизведения. В результате, выполняется воспроизведение сценария 70. После воспроизведения списка 73F воспроизведения снова отображаться может экран главного меню (например, предыдущий экран 60 меню).

Когда на экране 80А делают второй выбор, после отображения экрана 80А, воспроизводится список 73G воспроизведения. В списке воспроизведения 73G установлена метка в определенные моменты синхронизации. Когда воспроизводится

список 73G воспроизведения, список 73G воспроизведения может быть разделен в положении метки или может воспроизводиться полностью, в соответствии с установками устройства воспроизведения, другим сценарием пользователя или в соответствии с выбором на экране выбора разветвления. После воспроизведения всего списка 73G воспроизведения, последовательно воспроизводят списки 73М и 73I воспроизведения. После этого воспроизводят список 73J воспроизведения.

Когда список 73G воспроизведения разветвляется в положении метки, последовательно воспроизводят списки 73 К и 73L воспроизведения. После воспроизведения списка 73L воспроизведения, воспроизведение продолжают от положения метки, которая была установлена в списке 73I воспроизведения.

В списке 73J воспроизведения считываются графические данные 72В. При этом отображается экран 80В, на котором для пользователя представляется предложение выбрать раздел текстового элемента. При первом выборе на экране 80В воспроизводится список 73F воспроизведения. При втором выборе, на экране 80В воспроизводится список 73К воспроизведения от положения отметки, которая была установлена в списке 73К воспроизведения.

При воспроизведении сценария выполняются операции, соответствующие детектируемым меткам, вводимым командам пользователя, и изменения операций устройств воспроизведения в соответствии с последовательностями команд (программ), выполняемых устройством воспроизведения для списков воспроизведения.

Когда воспроизводится любой из списков воспроизведения 73А - 73М, при нажатии на кнопку меню пункта дистанционного управления отображается экран 60 меню для таблицы сценариев. Ниже будет описана работа, выполняемая в процессе воспроизведения списка воспроизведения для экрана 60 меню. В этом случае обработчик события, который соответствует событию, происходящему, когда нажимают на кнопку меню пульта дистанционного управления (событие нажатия на кнопку меню), и которое представляет собой команду, обеспечивающую возможность обработки списка воспроизведения для экрана 60 меню, описан как глобальный обработчик 71 события.

Один сценарий определен в директории. Один сценарий состоит из одного или множества списков воспроизведения. Директория представляет собой, например, вышеуказанную директорию BDAV стандарта записи и воспроизведения данных, и директорию HDMV (которая подробно будет описана ниже), которая, как предполагается, представляет собой область данных, предназначенных только для воспроизведения, в соответствии с вариантом выполнения настоящего изобретения.

Категории сценариев будут описаны со ссылками на фиг. 22, фиг. 23А, фиг. 23В и

на фиг. 23С. Основываясь на связях списков воспроизведения, структуры сценариев могут быть в основном разделены как три типа, которые представляют собой (1) одиночный список воспроизведения, (2) последовательный список воспроизведения и (3) список многократного воспроизведения, как показано на фиг. 22.

Одиночный список воспроизведения, обозначенный как (1), представляет собой сценарий, состоящий из одного списка воспроизведения, как показано на фиг. 23А. Для одиночного списка воспроизведения может быть определена временная ось. При этом перерыв во время воспроизведения сценария не предусмотрен. Когда содержание одиночного списка воспроизведения представляет собой кинофильм, после загрузки диска воспроизводится только основная часть кинофильма.

Последовательный список воспроизведения, обозначенный как категория (2), представляет собой сценарий, состоящий из множества списков воспроизведения, которые расположены линейно без разветвлений, как показано на фиг. 23В. Списки воспроизведения расположены таким образом, что конец одного списка воспроизведения соединен с началом следующего списка воспроизведения. В последовательном списке воспроизведения временная ось может быть определена для каждого списка воспроизведения. Когда содержание последовательного списка воспроизведения представляет собой кинофильм, сценарий состоит из экрана меню и основной части кинофильма. После загрузки диска выполняется список воспроизведения, который обеспечивает отображение экрана меню. Когда на экране меню обозначают воспроизведение основной части кинофильма, выполняется следующий список воспроизведения, и воспроизводится основная часть кинофильма.

Многократный список воспроизведения, обозначенный как категория (3), представляет собой сценарий, который содержит разветвление списка воспроизведения и соединение списков воспроизведения, как показано на фиг. 23С. В многократном списке воспроизведения временная ось не может быть определена через все списки воспроизведения. Вместо этого временная ось определена в каждом списке воспроизведения. При использовании многократного списка воспроизведения выполняется интерактивная функция и игровая функция, предназначенная для изменения содержания воспроизведения в соответствии с командами, вводимыми пользователем. Когда содержание многократного списка воспроизведения представляет собой кинофильм, может быть выполнена функция отображения под множеством углов обзора, которая позволяет пользователю выбирать требуемый угол из множества различных углов съемки одной и той же сцены.

В среде, предназначенной только для воспроизведения, один сценарий определен



для директории HDMV. Однако для пользователя необходимо обеспечить возможность рассматривать сценарий в виде более мелких модулей. Тем не менее, модуль списка воспроизведения не всегда соответствует модулю, который может рассматривать пользователь. Когда один список воспроизведения описывает три кинофильма, для пользователя необходимо обеспечить возможность видеть точку поиска каждого кинофильма. Точка поиска (точка входа), которая не зависит от структуры списка воспроизведения, называется заголовком и/или главой.

Далее, со ссылкой на фиг. 24, будут описаны заголовки и главы. Заголовок представляет собой любую начальную точку воспроизведения в сценарии. В примере, показанном на фиг. 24, заголовок 1 расположен в начале списка 470A воспроизведения. Заголовок 2 расположен в середине списка 470D воспроизведения. Область, следующая после начала списка 470A воспроизведения до заголовка 2 представляет собой заголовок 1. Глава представляет собой модуль, на который дополнительно подразделяют заголовок. Главу также можно рассматривать как точку начала воспроизведения. Заголовок 1 разделен на главы. В примере, показанном на фиг. 24, заголовок 1 содержит главы 1, 2 и 3. При этом заголовок 1 разделен на три участка. Как показано на фиг. 24, заголовки и главы могут быть расположены в середине списка воспроизведения.

Далее будет рассмотрено устройство воспроизведения, которое работает в соответствии с описанием сценария. Моделируемое устройство воспроизведения обозначается как виртуальное устройство воспроизведения ДГ (диска типа голубой луч). Определение структуры виртуального устройства воспроизведения ДГ называется моделью виртуального устройства воспроизведения ДГ.

Далее со ссылкой на фиг. 25 будет описана модель виртуального устройства воспроизведения ДГ. После загрузки диска в виртуальное устройство 30 воспроизведения ДГ, оно считывает с диска в качестве программы 40 УВП (РВС - управление воспроизведением), сценарий, описанный на языке описания сценария, определенном в настоящем изобретении, и работает в соответствии с описанием сценария.

Виртуальное устройство 30 воспроизведения ДГ воспроизводит данные с носителя записи в форме диска, который определен в соответствии с вариантом выполнения настоящего изобретения. Виртуальное устройство 30 воспроизведения ДГ представляет собой объект в компьютерной среде, такой, как персональный компьютер. Компьютерная среда не ограничивается персональным компьютером общего назначения. Вместо этого компьютерная среда включает программную среду, установленную на соответствующее устройство воспроизведения и/или устройство записи и воспроизведения, которое воспроизводит данные с носителя записи в форме диска, определенного в соответствии с

вариантом выполнения настоящего изобретения. Ниже носитель записи в форме диска, определенный в соответствии с вариантом выполнения настоящего изобретения, называется диском.

5            Виртуальное устройство 30 воспроизведения ДГ, в общем, имеет два состояния А и В. В состоянии А виртуальное устройство 30 воспроизведения ДГ воспроизводит список воспроизведения и графические изображения. В состоянии В виртуальное устройство 30  
10 воспроизведения ДГ прекращает воспроизведение списка воспроизведения и графических изображений. Изменение из одного состояния в другое состояние и определение следующей операции в одном состоянии выполняют по командам для объекта виртуального устройства 30 воспроизведения ДГ.

15            Состояние А содержит множество операций. В качестве операции в состоянии А принято воспроизведение с высокой скоростью, воспроизведение с переменной скоростью, например, воспроизведение в обратном порядке и специальное  
20 воспроизведение, например, воспроизведение с переходом, которое начинается с любого момента времени на диске. Когда отображаются данные плана 12 графического изображения, ограничивается воспроизведение с переменной скоростью и специальное воспроизведение виртуального устройства 30 воспроизведения ДГ.

25            Программа 40 УВП (управления воспроизведением) соответствует сценарию, записанному на диск. Как будет описано ниже, сценарий описывает способ воспроизведения для списка воспроизведения, записанного на диск, и способ отображения  
30 для экрана меню. Программа 40 УВП и виртуальное средство 30 воспроизведения ДГ выполняет обмен командами через API (интерфейс программ приложения) 41 для воспроизведения списка воспроизведения, записанного на диск.

35            Сценарий содержит две области для команд, включающих программу, содержащую команды, которые обеспечивают работу устройства воспроизведения. Эти две области обозначены как область глобальной команды и область локальной глобальной команды.  
40

          Область глобальной команды содержит программы, которые выполняются для всего сценария. Например, область глобальной команды описывает программу, которая обеспечивает инициализацию параметров устройства воспроизведения при загрузке диска  
45 в устройство воспроизведения и переход к списку воспроизведения, который составляет экран меню. Область локальной команды описывает программы для списков воспроизведения. Локальные команды разделены на четыре типа команд, которые представляют собой предварительные команды, команды элемента воспроизведения,  
50 последующие команды и команды кнопок.

На фиг. 26А и на фиг. 26В показана схема выполнения операций виртуального устройства 30 воспроизведения ДГ со сценарием, который описывает исходные команды в виде описательного языка в соответствии с вариантом выполнения настоящего изобретения. На фиг. 26А показан пример выполнения операции загрузки диска виртуального устройства 30 воспроизведения ДГ. Как описано выше, один сценарий создан для директории BDMV, который будет описан ниже. Когда диск загружают в устройство воспроизведения и затем выполняют исходный доступ к диску (на этапе S30), инициализируются регистры, а именно, общие параметры 32 (на этапе S31). На следующем этапе S32 программа считывается с диска и выполняется. Исходный доступ представляет собой операцию, в соответствии с которой воспроизведение диска выполняют в первый раз, например, когда диск загружают в устройство воспроизведения.

Группа команд (программа), которые первоначально считываются и выполняются при загрузке диска в устройство воспроизведения, называется глобальными командами. Глобальные команды описывают, например, изображение рекламы (рекламный анонс), и команду перехода, в соответствии с которой выполняется переход к списку воспроизведения, который составляет экран меню. Устройство воспроизведения воспроизводит список воспроизведения в соответствии с командами.

На фиг. 26В показан пример работы устройства 30 воспроизведения, когда пользователь нажимает, например, на клавишу воспроизведения, в то время как устройство воспроизведения находится в состоянии остановки. Эта операция соответствует изменению состояния из состояния В в состояние А виртуального устройства 30 воспроизведения ДГ, как описано на фиг. 25. Из состояния остановки (на этапе S40), пользователь переводит виртуальное устройство 30 воспроизведения ДГ в режим воспроизведения с помощью, например, пульта дистанционного управления (ОПД, UOP - операция пользователя). В результате, инициализируются регистры (а именно, общие параметры 32) (на этапе S41). На следующем этапе S42 виртуальное устройство 30 воспроизведения ДГ входит в фазу воспроизведения списка воспроизведения.

Далее, со ссылкой на фиг. 27А и на фиг. 27В, будет описано воспроизведение списка воспроизведения на фазе воспроизведения списка воспроизведения. На фиг. 27А показан пример, в котором список воспроизведения состоит из одиночного элемента воспроизведения. Список воспроизведения имеет область предварительной команды, область команды элемента воспроизведения и область последующей команды, которая описывает соответствующие программы. В фазе воспроизведения списка воспроизведения выполняется предварительная команда из области предварительной команды (на этапе S10). После выполнения предварительной команды устройство воспроизведения входит в

фазу воспроизведения элемента воспроизведения для воспроизведения элементов, которые составляют список воспроизведения (на этапе S11). В фазе воспроизведения элемента воспроизведения, воспроизводятся поток, начальная точка которого и конечная точка которого обозначены элементом воспроизведения (на этапе S110). После воспроизведения потока до конечной точки, выполняется команда элемента воспроизведения (на этапе S111). После выполнения команды элемента воспроизведения, выполняется последующая команда из области последующей команды (на этапе S12). В результате, воспроизводится список воспроизведения.

Последующая команда обычно представляет собой команду перехода, которая описывает, в качестве команды перехода, список воспроизведения, который воспроизводится следующим или список воспроизведения, который составляет экран меню. Когда команда перехода отсутствует, устройство воспроизведения входит в состояние остановки (состояние В, показанное на фиг. 25В).

На фиг. 27В показан пример, в котором список воспроизведения описывает множество элементов воспроизведения. В этом случае список воспроизведения содержит область предварительной команды, область команды элемента воспроизведения и область последующей команды, которые описывают соответствующие команды. Когда список воспроизведения описывает множество элементов воспроизведения, область команды элемента воспроизведения описывает потоки элемента воспроизведения и команды элемента воспроизведения для элементов воспроизведения, расположенных во временной последовательности.

Когда список воспроизведения описывает множество элементов воспроизведения, в фазе воспроизведения списка воспроизведения выполняется предварительная команда (на этапе S10). В фазе воспроизведения следующего элемента воспроизведения, поток воспроизводится от начальной точки до конечной точки каждого элемента воспроизведения, и команда элемента воспроизведения выполняется для каждого элемента воспроизведения. В примере, показанном на фиг. 27В, воспроизводится поток первого элемента воспроизведения (на этапе S110-1). Затем выполняется соответствующая команда элемента воспроизведения (на этапе S111-1). После этого воспроизводится поток второго элемента воспроизведения (не показан) (на этапе S110-2). При этом выполняется соответствующая команда элемента воспроизведения (на этапе S111-2). Эти операции повторяются для множества элементов воспроизведения. После воспроизведения потока последнего элемента воспроизведения (на этапе S110-n) и выполнения соответствующей команды элемента воспроизведения (на этапе S111-n), фаза воспроизведения элемента воспроизведения заканчивается. После окончания фазы

воспроизведения элемента воспроизведения выполняется следующая команда (на этапе S12). В результате, выполняется фаза воспроизведения списка воспроизведения.

5 На фиг. 28А, на фиг. 28В, на фиг. 28С, на фиг. 28D, фиг. 28Е, фиг. 28F, фиг. 28G и на фиг. 28H показана часть команд. Когда команды, примеры которых показаны на фиг. 28А, фиг. 28В, фиг. 28С, фиг. 28D, фиг. 28Е, фиг. 28F, фиг. 28G и на фиг. 28H описаны в области последующей команды и области команды кнопки, которая будет описана ниже, 10 может быть выполнен переход к заранее определенному списку воспроизведения. Помимо этих команд, показанных на фиг. 28А, фиг. 28В, фиг. 28С, фиг. 28D, фиг. 28Е, фиг. 28F, фиг. 28G, и фиг. 28H, могут быть определены другие команды.

15 Далее будут описаны команды, предназначенные для определения начального положения воспроизведения. Команда LinkPlayList (playListNumber) обеспечивает возможность воспроизведения из начинающегося списка воспроизведения, определяемого командой "playListNumber". Команда LinkPlayItem (playListNumber, playItemNumber) 20 обеспечивает возможность воспроизведения обозначенного элемента воспроизведения из обозначенного начинающегося списка воспроизведения. "playItemNumber" представляет собой идентификатор элемента воспроизведения "PlayItem\_id", значение которого начинается с "0". Когда "playItemNumber" определен равным "0", список воспроизведения, 25 к которому принадлежит данный элемент воспроизведения, воспроизводят с начала.

Команда Link (position) (object) обеспечивает перемещение текущего положения в сценарии. Эта команда обеспечивает перемещение текущего положения в соседний 30 список воспроизведения, элемент воспроизведения или главу. Параметр "position" (положение) представляет собой один из "prev", "next", "top", "Parent" или "tail" ("предыдущий", "следующий", "верхний", "родительский" или "конечный"). Параметр 35 "object" (объект) описывает способ перемещения для объекта (списка воспроизведения, элемента воспроизведения или главы), представленного параметром "object".

Команда Exit (выход) обеспечивает остановку воспроизведения сценария. В этом случае значение стандартного регистра не сохраняется. Команда RSM обеспечивает вызов 40 информации продолжения, хранящейся в запоминающем устройстве воспроизведения, установку ее в регистр и начало воспроизведения сценария.

Ниже будут описаны команды, предназначенные для получения состояния 45 устройства воспроизведения. Команда getMenuDescriptionLanguage() обеспечивает получение языка, используемого для отображения меню. Команда getScenarioNumber(), команда getPlayListNumber() и команда getChapterNumber() обеспечивают получения соответственно воспроизводимых номера сценария, номера списка воспроизведения и 50 номера главы. Команда getPlayerSupport() обеспечивает получение информации - версии

устройства воспроизведения.

Далее будут описаны команды для видеопотоков. Команда  
5 `getVideoStreamAvailability()` обеспечивает получение информации, которая описывает,  
действительно ли должен быть включен обозначенный видеопоток или нет. Команда  
`setVideoStreamNumber()` описывает декодируемый видеопоток. Команда  
`getVideoStreamNumber()` обеспечивает получение номера выбранного видеопотока.  
10 Атрибуты видеопотока представляют собой, например, систему кодирования,  
разрешающую способность, соотношение размеров, режим визуального отображения в  
случае, когда соотношение размеров составляет - 4: 3, и присутствие/отсутствие закрытых  
15 надписей. Команда `setAngleNumber()` описывает номер угла. Команда `getAngleNumber()`  
обеспечивает получение выбранного номера угла. Команда `getMaxVideoStreams()`  
обеспечивает получение максимального количества битовых потоков.

Далее, в соответствии с вариантом выполнения настоящего изобретения, будет  
20 описан способ записи на диск команд и баз данных, которые описывают сценарий. На  
фиг. 29 показан пример структуры управления файлами в соответствии с вариантом  
выполнения настоящего изобретения. На диске создана одна корневая директория.  
25 Участок в пределах корневой директории управляется с помощью одной системы  
воспроизведения.

В корневой директории расположена директория BDMV. Хотя здесь это не  
показано, аналогично фиг. 4, описанной выше, в корневой директории может быть  
30 расположено множество директорий BDMV. Ниже система, определенная в варианте  
выполнения настоящего изобретения, называется BDMV.

В директории BDMV расположены два файла "scenario.hdmv" и "entrylist.data".  
35 Кроме того, здесь расположено множество директорий "PLAYLIST", "CLIPINF" и  
"STREAM".

На фиг. 30 показан синтаксис, который описывает пример структуры файла  
40 "scenario.hdmv". Файл "scenario.hdmv" представляет собой файл, который первоначально  
считывают и выполняют при выполнении исходного доступа (а именно при загрузке  
диска). Файл "scenario.hdmv" имеет код идентификации файла (поле `type_indicator`) и  
номер версии (поле `version_number`). После номера версии следуют блоки  
45 функциональных данных.

Поле `type_indicator` имеет длину данных 32 бита. Поле `type_indicator` имеет  
заданную строку знаков, которая описывает, что файл представляет собой "scenario.hdmv".  
50 Поле `version_number` имеет длину данных 32 бита и предназначено для номера версии.  
Поле `Scenario_start_address` имеет длину данных 32 бита и предназначено для

целочисленного значения без знака, которое представляет положение блока Scenario() с относительным количеством байтов от начала файла "scenario.hdmv".

5 Блок Autoplay() начинается с 41-ого байта (фиксированное положение) файла. Блок Autoplay() описывает программу, которая выполняется, при выполнении исходного доступа (первоначальное воспроизведение диска, выполняемое, например, при загрузке диска). Блок Autoplay() содержит приведенные выше глобальные команды, описанные  
10 выше. После блока Autoplay() следует любое количество слов заполнения (padding\_word), что позволяет сформировать пространство после блока.

На фиг. 31 показан синтаксис, который описывает пример структуры блока Autoplay(), показанного на фиг. 30. Поле length имеет длину данных 32 бита для целого  
15 числа без знака. Поле length описывает длину в байтах данных, следующих непосредственно после окончания поля length до окончания блока Autoplay(). Поле number\_of\_commands описывает количество полей command(i), которые следуют после него. Поле command(i) имеет длину данных 32 бита, как показано на фиг. 28А, фиг. 28В,  
20 фиг. 28С, фиг. 28D, фиг. 28Е, фиг. 28F, фиг. 28G и фиг. 28Н. Поле command(i) описывает набор параметров устройства воспроизведения и команд таких, как команды начала воспроизведения для назначенного списка воспроизведения и команды расчета.  
25

Блок Scenario() описывает "сценарий" как описано выше. Блок Scenario() описывает информацию о порядке воспроизведения списков воспроизведения и области локальной  
30 команды для каждого списка воспроизведения.

На фиг. 32 показан синтаксис, который описывает пример структуры блока Scenario(). Блок Scenario() представляет собой блок, который определяет информацию о сценарии, а именно связи между списками воспроизведения. Блок Scenario() описывает  
35 информацию о предыдущих предварительных командах, последующих командах и командах элемента воспроизведения и сами эти команды. Блок Scenario() содержит область для полей информации о командах, описанных в нем, и область для этих предварительных команд, последующих команд и команд элемента воспроизведения.  
40

Поле length описывает значение, которое представляет длину непосредственно после окончания поля length до конца блока Scenario() в байтах. Поле number\_of\_PlayLists описывает количество списков воспроизведения, которое составляет сценарий. После  
45 поля number\_of\_PlayLists следуют данные каждого списка воспроизведения. Данные для каждого списка воспроизведения повторяются количество раз, определяемое счетчиком  $i$  циклов в цикле for, причем максимальное значение  $i$  представлено как поле number\_of\_PlayLists.  
50

Поле Pre\_Command\_start\_id описывает начальный номер предварительной

команды в таблице команд (предварительная команда выполняется перед воспроизведением списка воспроизведения). Номер, описанный в поле Pre\_Command\_start\_id, описывает счетчик j циклов в цикле for, который описывает поле PI\_Command (i), которое будет описано ниже. Аналогично, поле Post\_Command\_start\_id описывает начальный номер последующей команды в таблице команд (последующая команда выполняется после воспроизведения списка воспроизведения). Номер, описанный в поле Post\_Command\_start\_id, описывает счетчик j циклов в цикле for, который описывает поле PL \_\_ Command (j), которое будет описано ниже.

Поле Number\_of\_Pre\_Commands описывает количество предварительных команд, которые составляют программу. Предварительные команды выполняются перед воспроизведением списка воспроизведения. Аналогично, поле number\_of\_Post\_Commands описывает количество последующих команд, которые составляют программу. Последующие команды выполняются после воспроизведения списка воспроизведения. Эти команды описаны в таблице команд, которая будет описана ниже.

Поле number\_of\_PlayItems описывает количество элементов воспроизведения, которые составляют элемент воспроизведения. Поле PI\_Command\_start\_id представляет начальный номер команды элемента воспроизведения в таблице команд. Команда элемента воспроизведения выполняется после воспроизведения элемента воспроизведения. Число, описанное в поле PI\_Command\_start\_id, описывает счетчик j циклов в таблице команд, которая будет описана ниже. Поле number\_of\_PI\_Commands описывает количество команд элемента воспроизведения, которые выполняются после воспроизведения элемента воспроизведения. Команды после положения, описанного в поле PI\_Command\_start\_id, до номера команд, описанных в поле number\_of\_PI\_Commands, выполняются после воспроизведения элемента воспроизведения.

Поле number\_of\_PL\_Commands описывает количество команд в таблице команд, которая следует после поля number\_of\_PL\_Commands. Таблица команд содержит цикл for, который описывает поле PL\_Command (j). Командам в таблице команд назначают номер j. Номер j соответствует счетчику j циклов в цикле for, который описывает таблицу команд. Поле PL\_Command (j) описывает одну команду. Ссылка на номер j делается из предыдущего поля Pre\_Command\_start\_id, поля Post\_Command\_start\_id и поля PI\_Command\_start\_id.

На фиг. 33 показан синтаксис, который описывает пример структуры данных файла "entrylist.data". Файл "entrylist.data" описывает идентификационный код файла (поле type\_indicator), номер версии (поле version\_number) и начальный адрес блока (поле ScenarioEntry\_start\_address). После начального адреса блока следуют блоки



функциональных данных.

Поле `type_indicator` имеет длину данных 32 бита для заданной строки знаков, которая описывает входные точки заголовка и меню. Поле `version_number` имеет длину  
5 данных 32 бита для номера версии. Поле `ScenarioEntry_start_address` имеет длину данных 32 бита для целого значения без знака, которое описывает начальное положение блока `ScenarioEntry()` с относительным количеством байтов от начала поля `"entrylist.data"`.

На фиг. 34 показан синтаксис, который описывает пример структуры блока `AppInfo()`. Поле `length` имеет длину данных 32 бита для целого числа без знака, которое описывает длину немедленно после окончания поля `length` до конца блока `AppInfo()` в  
10 байтах. Поле `HDMV_name_character_set` описывает набор знаков для поля `HDMV_name`, которое будет описано ниже. Поле `PIN_valid_flag` описывает, действительно ли личный идентификационный номер должен быть установлен при воспроизведении. Когда эта установка разрешена, поле `PIN`, после которого следует поле `PIN_valid_flag`, `rjnjht`  
15 описывает личный идентификационный номер. Поле `HDMV_name_length` описывает длину разрешенного участка поля `HDMV_name`, которое следует после поля `HDMV_name_length`. Поле `HDMV_name` представляет собой область, которая описывает название директории `HDMV`, в которой расположен файл `"entrylist.data"`. Название директории `HDMV` описано в текстовом формате. Поле `HDMV_name` имеет фиксированную длину данных 255 байтов. Название директории `HDMV` описано для  
20 длины, описанной в поле `HDMV_name_length` после начала поля `HDMV_name`.

На фиг. 35 показан синтаксис, который описывает пример структуры блока `ScenarioEntry()`. Блок `ScenarioEntry()` описывает точки поиска сценария. Как описано  
30 выше, сценарий создается для директории `HDMV`. Сценарий соединяет множество списков воспроизведения, размещенных в директории `HDMV`, для определения порядка воспроизведения списков воспроизведения. Когда пользователь видит сценарий, он не всегда выглядит как один изобразительный модуль или один звуковой модуль, но так, как  
35 если бы он состоял из множества "заголовков".

Когда на один диск записаны три кинофильма, на диске имеется только один сценарий, который определяет порядок воспроизведения кинофильмов. Однако  
40 пользователь может их смотреть, как если бы три заголовка были записаны на диск. В качестве альтернативы, может отображаться список из трех заголовков. Благодаря включению меню заголовка, которое позволяет пользователю выбирать один из заголовков, он или она может видеть их, как если бы были записаны четыре заголовка.  
45 Поскольку пользователь рассматривает экран меню как один изобразительный модуль или один звуковой модуль, в соответствии с данным вариантом выполнения настоящего  
50

изобретения, экран меню обрабатывается как один тип заголовка.

Поскольку модуль сценария, который определяет связь списков воспроизведения, отличается от модуля, в котором пользователь распознает данные как изображение и звук, как описано выше, требуется определить точки поиска в сценарии. Точка поиска в сценарии называется входом заголовка. Блок ScenarioEntry() описывает информацию входа заголовка.

Возвращаясь к фиг. 35, можно видеть, что поле length имеет длину данных 32 бита для целого числа без знака, которое описывает длину непосредственно после поля length до окончания блока ScenarioEntry() в байтах. Поле name\_character\_set описывает набор знаков поля TopMenu\_name и поля Title\_name, которые следуют после поля name\_character\_set.

Следующий блок Top menu PL() описывает точку входа в список воспроизведения или группу списка воспроизведения, которая составляет меню, отображаемое, когда пользователь нажимает на клавишу меню заголовка или на пульт дистанционного управления. Один сценарий содержит одно верхнее меню. Верхнее меню используется для представления пользователю, например, заголовков. Подменю, в которых пользователь может устанавливать звуковое сопровождение и подзаголовки, может быть установлено как меню, расположенное ниже верхнего меню. Подменю также называется меню установки потока.

Вкратце, поле flags (флаги) представляет собой область, которая описывает информацию атрибута верхнего меню. Поле TopMenu\_ref\_to\_PlayList\_file\_name описывает список воспроизведения, который составляет верхнее меню или список воспроизведения, который представляет собой вход в группу списка воспроизведения. Поле TopMenu\_ref\_to\_PlayItem\_id описывает количество элементов воспроизведения, от которых начинается верхнее меню в списке воспроизведения, описанном в поле TopMenu\_ref\_to\_PlayList\_file\_name. Когда список воспроизведения воспроизводят с начала, значение поля TopMenu\_ref\_to\_PlayItem\_id устанавливают равным "0". Поле TopMenu\_name\_length представляет длину названия, назначенного верхнему меню. Поле TopMenu\_name описывает строку знаков названия, назначенного верхнему меню.

Блок Top Menu PL() описывает информацию о заголовке. Поле number\_of\_Titles описывает количество точек поиска заголовка (входов заголовка) в цикле for, который следует сразу после поля number\_of\_Titles. Вкратце, поле flags представляет собой область, которая описывает информацию атрибута о заголовке. Поле Title\_ref\_to\_PlayList\_file\_name описывает название файла списка воспроизведения, который включает вход в заголовок. Поле Title\_ref\_to\_PlayItem\_id используется, когда

заголовок начинается с определенного элемента воспроизведения списка воспроизведения, описанного в поле `Title_ref_to_PlayList_file_name`. Поле `Title_name_length` описывает длину названия, назначенного заголовку. Поле `Title_name` описывает строку знаков названия, назначенного заголовку.

Далее описана информация о подменю. После "Меню установки потока" следует точка входа в список воспроизведения или группу списка воспроизведения, которая составляет меню установки потока (а именно, подменю) для каждого элемента воспроизведения. Меню установки потока можно использовать для каждого списка воспроизведения, для выбора, например, звука, подзаголовков или угла обзора. Например, когда нажимают на кнопки 64 и 65, показанные на фиг. 20, подменю отображается как экран.

Поле `number_of_PlayLists` описывает количество списков воспроизведения, используемых для меню установки потока. Значение поля `number_of_PlayLists` используется как количество циклов `for`, следующих сразу же после поля `number_of_PlayLists`. Вкратце, поле `SSMenu_flags` представляет собой область, которая описывает информацию атрибута о меню установки потока. Поле `SSMenu_ref_to_PlayList_file_name` описывает список воспроизведения, который составляет меню установки потока или список воспроизведения, который представляет собой вход в группу списка воспроизведения. Поле `SSMenu_ref_to_PlayItem_id` описывает количество элементов воспроизведения, от которых начинается меню установки потока в списке воспроизведения, описанном в поле `SSMenu_ref_to_PlayList_file_name`. Когда список воспроизведения воспроизводят с начала, значение поля `SSMenu_ref_to_PlayItem_id` равно "0".

На фиг. 36 показан синтаксис, который описывает пример структуры файла "xxxxx.mpls". На фиг. 36 файл "xxxxx.mpls" содержит функциональные блоки информации. Поле `type_indicator` описывает строку знаков файла. Поле `version_number` описывает версию файла. Поля `PlayList_start_address` и `PlayListMark_start_address` описывают начальные адреса соответствующих блоков в виде информации адреса, имеющей длину данных 32 бита.

Блок `PLControlInfo()` описывает информацию атрибута о списке воспроизведения. Блок `PlayList()` описывает информацию об элементе воспроизведения, который составляет список воспроизведения. Блок `PlayListMark()` описывает информацию о метке, добавленной к списку воспроизведения.

В файле "xxxxx.mpls", поскольку после начальных адресов блока `PLControlInfo()`, `PlayList()` и `PlayListMark()` следуют эти блоки, данные заполнения `padding_word` могут

быть расположены перед и/или после каждого блока с требуемой длиной. Начальное положение первого блока PLControlInfo() фиксировано на 41-ом байте от начала файла.

5 На фиг. 37 показан синтаксис, который описывает пример структуры блока PLControlInfo(). Блок PLControlInfo() описывает различные типы информации атрибута о списке воспроизведения, не требующейся непосредственно для воспроизведения списка воспроизведения. Поле Playlist\_character\_set описывает набор знаков информации строки 10 знаков о списке воспроизведения.

Поле PL\_playback\_type описывает значение, как показано на фиг. 38. Поле PL\_playback\_type описывает, действительно ли список воспроизведения представляет собой обычный последовательно воспроизводимый список воспроизведения, список 15 воспроизведения элементы воспроизведения которого воспроизводятся в случайном порядке, или список воспроизведения элементы воспроизведения которого воспроизводятся после перестановки. Случайная перестановка определяется в модуле списка воспроизведения. Один список воспроизведения не должен описывать регулярно воспроизводимый элемент воспроизведения и блок элемента воспроизведения со 20 случайной перестановкой.

25 Когда диск представляет носитель записи, предназначенный только для воспроизведения, производитель может определять случайное воспроизведение или воспроизведение с перестановкой. В этот момент времени такая информация необходима.

Поле playback\_count описывает количество раз воспроизведения элемента 30 воспроизведения, когда список воспроизведения представляет собой список воспроизведения со случайным воспроизведением или список воспроизведения с перестановкой. Поле playback\_count описывает количество элементов воспроизведения, 35 которые воспроизводят со случайным порядком или воспроизводят с перестановкой.

Поле PL\_UOP\_mask\_table() описывает информацию об ограничениях операций 40 пользователя. Когда пользователю запрещено выполнять такие операции, как воспроизведение, быстрая перемотка вперед, быстрая перемотка назад и т. д. при воспроизведении списка воспроизведения, эта область описана соответствующим образом. Когда соответствующее значение описано в поле PL\_UOP\_mask\_table(), 45 исключается возможность пропуска предупреждающего уведомления, уведомления о защите авторских прав и т. д., даже если выполняется операции быстрой перемотки вперед или аналогичная операция.

Поле PL\_random\_access\_mode описывает значение, показанное на фиг. 39. Поле 50 PL\_random\_access\_ описывает, может ли быть выполнен случайный доступ к любому положению списка воспроизведения с переходом. Когда существует список

воспроизведения, который производитель желает представить пользователю, значение поля `PL_random_access_mode` устанавливаются в  $[0 \times 1]$ . Когда этот список воспроизведения воспроизводят с переходом, операция быстрой перемотки вперед, операция быстрой перемотки назад, воспроизведение из любой точки времени и т. д. запрещены. Когда диск представляет собой носитель записи, предназначенный только для воспроизведения, на него могут быть записаны сцены, такие, как логотип компании - производителя содержания, и предупреждения, которые должны быть представлены пользователю. Поле `PL_random_access_mode` описывает информацию, необходимую для запрета пропуска таких сцен при операции воспроизведения с переменной скоростью или тому подобное.

Поле `PlayList_duration` описывает длительность воспроизведения списка воспроизведения. Поле `PlayList_name` описывает название списка воспроизведения, имеющего эффективную длину со значением, описанным в поле `PlayList_name_length`. Поле `PlayList_detail` описывает подробную информацию о списке воспроизведения, имеющем эффективную длину со значением, описанным в поле `PlayList_detail_length`.

На фиг. 40 показан синтаксис, который описывает пример структуры блока `PlayList()`. Поле `length` описывает длину в байтах непосредственно после поля `length` до конца блока `PlayList()`. Поле `number_of_PlayItems` описывает количество элементов воспроизведения, которые составляют список воспроизведения. Поле `number_of_SubPlayItems` описывает количество вспомогательных элементов воспроизведения (подэлементов воспроизведения), воспроизводимых наряду с основным элементом воспроизведения.

Блок `PlayItem()` описывает информацию об элементе воспроизведения. Блок `SubPlayItem()` описывает информацию о подэлементе воспроизведения.

На фиг. 41 показан синтаксис, который описывает пример структуры блока `PlayItem()`. Поле `Clip_Information_file_name` описывает строку знаков названия файла информации клипа (который имеет расширение `"clpi"`), соответствующего клипу, на который ссылается элемент воспроизведения в соотношении 1 к 1.

Поле `Clip_codec_identifier` описывает систему кодирования клипа, на который ссылается элемент воспроизведения. В соответствии с вариантом выполнения, поле `Clip_codec_identifier` описывает фиксированное значение `"M2TS2"`. Другими словами, в соответствии с данным вариантом выполнения, система кодирования клипа, на которую ссылается элемент воспроизведения, представляет собой фиксированную систему, представленную значением `"M2TS"`.

Флаг `is_multi_angle` описывает, действительно ли элемент воспроизведения имеет структуру множества углов обзора (первый флаг).

Поле `connection_condition` представляет собой информацию, которая описывает способ соединения элемента воспроизведения и следующего элемента воспроизведения. Поле `connection_condition` описывает, действительно ли пункты воспроизведения могут воспроизводиться без стыков.

Поле `ref_to_STC_id` описывает последовательность `STC_sequence` в клипе, на который ссылается элемент воспроизведения. Последовательность `STC_sequence` имеет уникальную структуру стандарта диска типа "голубой луч", который описывает, что ТЧП (опорная тактовая частота программы) как опорная частота временной оси MPEG2 ТП (транспортного потока) представляет собой непрерывный диапазон. Последовательность `STC_sequence` описывает номер `STC_id`, который является уникальным в этом клипе. Поскольку в последовательности `STC_sequence` может быть определена непрерывная временная ось, время начала и время окончания пункта воспроизведения может быть уникально определено. Другими словами, начальная точка и конечная точка каждого элемента воспроизведения должна быть представлена в одной последовательности `STC_sequence`. Поле `ref_to_STC_id` описывает последовательность `STC_sequence` с номером `STC_id`.

Поля `IN_time` и `OUT_time` описывают временные штампы `pts` (`presentation_time_stamp`) начальной точки и конечной точки элемента воспроизведения в последовательности `STC_sequence`, соответственно.

Поле `PI_UOP_mask_table()` описывает данные об ограничениях операции пользователя. Даже если пользователь выполняет такую ограниченную операцию, устройство воспроизведения не должно отвечать на нее. Для ограничения операции быстрой перемотки вперед при отображении экрана меню, данные об ограничениях такой операции описаны в поле `PI_UOP_mask_table()`.

Поле `PI_UOP_mask_table()` описано для каждого элемента воспроизведения. Поле `PI_UOP_mask_table()` описывает информацию, которая имеет тот же объект, что и поле `PL_UOP_mask_table()` предыдущего блока `PLControlInfo()`, который описывает информацию о воспроизведении списка воспроизведения. Операция пользователя может быть запрещена в списке воспроизведения или в элементе воспроизведения. Операция пользователя при воспроизведении элемента воспроизведения запрещена в зависимости от результата операции ИЛИ информации списка воспроизведения и информации элемента воспроизведения.

Вкратце, поле `PID_filter()` представляет собой таблицу, которая описывает приоритет в потоках, воспроизводимых элементами воспроизведения.

Поле `PI_random_access_mode` описывает значение, представленное на фиг. 42. Поле

PI\_random\_access\_mode описывает, действительно ли может быть выполнен доступ в случайном порядке для воспроизведения с переходом для любого положения элемента воспроизведения. Когда существует список воспроизведения, который производитель диска желает показать пользователю, в поле PI\_random\_access\_mode описано значение [0 x 1]. Таким образом, при начале воспроизведения элемента воспроизведения, пользователю может быть запрещено выполнять операцию быстрой перемотки вперед, операцию обратной перемотки, операцию воспроизведения с любого момента времени или тому подобное.

Поле still\_mode описывает, действительно ли после начала воспроизведения элемента воспроизведения, воспроизведение временно останавливается. Поле still\_mode описывает значение, показанное на фиг. 43. Когда значение поля still\_mode равно [0 x 1], оно описывает, что воспроизведение элемента воспроизведения временно останавливается на период, описанный в следующем поле still\_time. При этом неподвижные изображения могут быть последовательно отображаться через интервалы с заданным периодом как при демонстрации слайдов. В этом случае каждое неподвижное изображение представляет собой элемент воспроизведения. Кроме установки определенного периода времени, в поле still\_time может быть описана установка неопределенного периода времени, когда воспроизведение останавливается до тех пор, пока пользователь не введет данные (установка паузы). Установка паузы может выполняться когда значение поля still\_mode равно [0 x 2].

Когда значение вышеуказанного флага is\_multi\_angle представляет собой, например, "1", элемент воспроизведения представляет собой элемент воспроизведения с множеством углов обзора. После "Angle" добавлена информация о множестве углов обзора.

Поле number\_of\_angles описывает количество углов обзора. Поле is\_seamless\_angle\_change описывает значение, представленное на фиг. 44. Поле is\_seamless\_angle\_change описывает, действительно ли каждый угол был записан на диск так, что каждый угол можно изменять без стыков.

Следующий цикл for описывает информацию о клипах, которые составляют углы. Поле Clip\_Information\_file\_name в цикле for описывает строку знаков названия поля файла информации клипа (который имеет расширение ".clri"), которое соответствует каждому клипу, на который ссылается элемент воспроизведения в соотношении 1 к 1. Поле ref\_to\_STC\_id описывает последовательность SC\_sequence для каждого клипа, на который ссылается этот элемент воспроизведения.

Угол, соответствующий значению angle\_id = 0, был определен в первой половине

блока `PlayItem()`, как нормальный элемент воспроизведения, который не представляет собой элемент воспроизведения по углом. Углы после значения `angle_id = 1` определены в цикле `for`. Цикл `for` не содержит угол, соответствующий значению `angle = 0`.

5 На фиг. 45 показан синтаксис, который описывает пример структуры блока `SubPlayItem()`. Поле `length` описывает длину в байтах непосредственно после поля `length` до окончания блока `SubPlayItem()`. Поле `Clip_Information_file_name` описывает строку  
10 знаков в названии файла информации клипа (который имеет расширение `"clpi"`), соответствующего клипу, на который ссылается подэлемент воспроизведения в соотношении 1 к 1.

15 Поле `Clip_codec_identifier` описывает систему кодирования клипа, на который ссылается подпункт воспроизведения. В соответствии с вариантом выполнения, поле `Clip_codec_Identifier` является фиксированным и содержит значение `"M2TS"`.

20 Поле `is_repeat_flag` описывает значение, показанное на фиг. 46. Поле `is_repeat_flag` представляет собой флаг, который описывает, действительно ли подэлемент воспроизведения воспроизводится многократно, не синхронно с основным элементом воспроизведения (основной путь). Когда значение поля `is_repeat_flag` равно `"1"`,  
25 подэлемент воспроизведения воспроизводится многократно до тех пор, пока основной элемент воспроизведения не будет воспроизведен не синхронно с ним. Когда значение поля `is_repeat_flag` равно `"0"`, подэлемент воспроизведения воспроизводится один раз, синхронно с основным элементом воспроизведения.

30 Если подэлемент воспроизведения представляет собой подэлемент воспроизведения только звука, когда в поле `is_repeat_flag` описана `"1"`, может воспроизводиться ФМС (BGM фоновое музыкальное сопровождение).

35 Поле `SubPlayItem_type` описывает характеристику, которую содержит подэлемент воспроизведения. Например, когда значение поля `SubPlayItem_type` равно `"1"`, оно описывает, что элемент воспроизведения представляет собой подэлемент  
40 воспроизведения только для звука.

45 Поле `ref_to_STC_id` описывает последовательность `STC_sequence` клипа, на который ссылается элемент воспроизведения. Поля `SubPlayItem_IN_time` и `SubPlayItem_OUT_Time` описывают штампы времени (`presentation_time_stamp`) начальной точки и конечной точки подэлемента воспроизведения в последовательности `STC_sequence`.

50 Когда значение предыдущего поля `is_repeat_flag` равно `"0"`, и оно представляет, что подэлемент воспроизведения воспроизводится синхронно с основным элементом воспроизведения, поле `sync_PlayItem_id` и поле `sync_start_PTS_of_PlayItem` описывают



момент времени, с которого подэлемент воспроизведения основного элемента воспроизведения воспроизводится синхронно с ним.

5 Как показано на фиг. 47, поле `sync_PlayItem_id` описывает элемент воспроизведения основного пути (`PlayItem = 1`). Поле `sync_start_PTS_of_PlayItem` описывает время ( $t_1$ ) основного элемента воспроизведения, в которое начинается воспроизведение подэлемента воспроизведения. Поле `SubPlayItem_IN_time` и поле `SubPlayItem_OUT_time` описывают период, в течение которого воспроизводится клип, как подэлемент воспроизведения.

10 На фиг. 48 показан синтаксис, который описывает пример структуры файла "zzzzz.clpi". На фиг. 48 показан файл "zzzzz.clpi", содержащий функциональные блоки информации. Поле `type_indicator` описывает строку знаков файла. Поле `version_number` описывает версию файла. Поле `SequenceInfo_start_address`, поле `ProgramInfo_start_address`, поле `CPI_start_address` и поле `ClipMark_start_address` описывают начальные положения соответствующих блоков.

15 На фиг. 49 показан синтаксис, который описывает пример структуры блока `ClipInfo()`. Поле `length` описывает длину сразу после поля `length` до окончания блока `ClipInfo()`. Поле `Clip_stream_type` описывает тип АВ потока клипа. Значение поля `Clip_stream_type` может быть фиксировано, например, равным "1", что представляет нормальный клип в стандарте диска, предназначенного только для воспроизведения.

20 Поле `application_type` описывает, вид мультиплексирования АВ потока клипа (который имеет расширение "m2ts"). Поле `application_type` описывает значение, показанное на фиг. 50. Поле `application_type` описывает, действительно ли АВ поток клипа представляет собой нормальный видеопоток или поток, который был соответствующим образом мультиплексирован для неподвижного изображения.

25 На практике, например, значение поля `application_type` равно "1", и это описывает то, что файл соответствующего АВ поток клипа соответствует правилу транспортного потока HDMV в соответствии с вариантом выполнения. При использовании АВ потока клипа воспроизводится обычное подвижное изображение.

30 Когда значение поля `application_type` равно "2", это описывает, что файл соответствующего АВ потока клипа соответствует правилу транспортного потока HDMV для неподвижного изображения, которое синхронизировано с воспроизведением звука. АВ поток клипа представляет собой файл в соответствии с, например, форматом MPEG2. АВ поток клипа содержит мультиплексированные видеоданные и аудиоданные. Видеоданные имеют структуру, в которой  $i$  изображений в формате MPEG2 расположены как неподвижные изображения. В результате, неподвижные изображения можно

воспроизводить в виде демонстрация слайдов на временной оси звука. Такое воспроизведение называется демонстрацией слайдов на временной оси.

5 Когда значение поля `application_type` равно "3", это описывает, что файл соответствующего АВ потока клипа соответствует правилу транспортного потока HDMV для неподвижных изображений, воспроизводимых не синхронно со звуком. Данные звука и видеоданные структурированы как различные файлы. При этом при воспроизведении 10 данных звука, видеоданные отображаются таким образом, что неподвижные изображения изменяются через любые интервалы или в соответствии с определением пользователя. Видеоданные могут быть структурированы таким образом, что, например, *i* изображений в формате MPEG2 располагаются как неподвижные изображения. Такое воспроизведение 15 называется как демонстрацией слайдов с возможностью произвольного просмотра.

Когда значение поля `application_type` равно "0", соответствующий АВ поток клипа не соответствует правилу транспортного потока HDMV.

20 Предполагается, что мультиплексирование, соответствующее отображению неподвижных изображений, позволяет легко выполнять такое программное приложение, как демонстрация слайдов неподвижных изображений. В таком варианте применения, 25 когда одно неподвижное изображение является капсулированно-мультиплексированным с подзаголовками и графическими данными, которые должны быть наложены на него, их можно легко считывать.

30 Когда неподвижное изображение мультиплексировано с подзаголовками и графическим изображением так же, как в обычном подвижном изображении, подзаголовки, отображаемые вместе с неподвижным изображением, мультиплексированы с данными изображения предыдущего неподвижного изображения (а именно, при этом 35 возникает, так называемая, разность фаз мультиплексирования). В результате, если только данные потока не будут считаны в течение длительного времени, неподвижное изображение, на которое наложены подзаголовки и графические изображения, не может 40 быть отображено.

В соответствии с вариантом выполнения настоящего изобретения, графические данные для видеоданных и подзаголовков содержатся в пакетах ТП (транспортного потока) MPEG. Один пакет ТП состоит из 188 байтов. Предыдущие видеоданные и 45 графические данные разделены так, что они содержатся в пакетах ТП. Когда после пакета данных подзаголовка, соответствующих определенным данным неподвижного изображения (называемого изображением Р1) следует пакет следующего неподвижного изображения (называемого изображением Р2), для отображения подзаголовков, 50 соответствующих изображению Р1, должны быть считаны данные изображения Р2.

Когда определенное неподвижное изображение мультимплексируется только с ассоциированными подзаголовками и графическим изображением (капсулировано), может быть сформирован поток, на который не влияют другие данные. Когда такая операция  
5 повторяется для каждого неподвижного изображения, и потоки соединены, может быть получен один поток, в котором последовательно соединены данные каждого неподвижного изображения (и ассоциированные данные подзаголовков и графические  
10 данные). Поток, мультимплексируемый таким образом, называется потоком HDMV неподвижного изображения.

Существует два типа потоков HDMV для неподвижных изображений, которые представляют собой демонстрацию слайдов на временной оси и демонстрацию слайдов с  
15 возможностью произвольного просмотра. В соответствии с вариантом выполнения, два типа отличаются по различным числам в поле `application_type`.

Когда неподвижное изображение и соответствующие подзаголовки и графические изображения капсулированы и записаны, изменяется возможность доступа к таким  
20 неподвижным изображениям и улучшается воспроизведение.

Возвращаясь к фиг. 49, тип поля `Clip_stream_` описывает тип АВ потока клипа. Поле `num_of_source_packets` описывает количество пакетов, содержащихся в АВ потоке  
25 клипа. Поле `BD_system_use` и блок `TS_type_info_block()` не относятся к настоящему изобретению. Их описание здесь не приведено.

На фиг. 51 показан синтаксис, который описывает пример структуры блока `SequenceInfo()`. Поле `length` описывает длину в байтах непосредственно после поля `length`  
30 до окончания блока `SequenceInfo()`. Поле `num_of_ATC_sequence` описывает количество последовательностей `ATC_sequence`, которые были записаны за непрерывный период времени. Когда используется носитель, предназначенный только для воспроизведения, поскольку количество последовательностей `ATC_sequence` равно "1", описание поля  
35 `num_of_ATC_sequences` не будет здесь приведено. Поле `SPN_ATC_start` описывает начало последовательности `ATC_sequence` с номером пакета. Когда количество последовательностей `ATC_sequence` равно "1", начало последовательности `ATC_sequence`  
40 соответствует началу файла АВ потока клипа. Таким образом, значение поля `SPN_ATC_start` равно "0".

Поле `num_of_STC_sequences` описывает количество последовательностей `STC_sequence` последовательности `ATC_sequence`. Когда используется носитель,  
45 предназначенный только для воспроизведения, поскольку количество последовательностей `STC_sequence` равно "1", описание его здесь не приводится. Поле `offset_STC_id` описывает фиксированное значение "0". Поле `PCR_PID` описывает ИДП  
50

(PID - идентификатор пакета) пакета ТП, который содержит ТЧП (опорную тактовую частоту программы) MPEG2 ТП. Поле SPN\_STC\_start описывает начало последовательности STC\_sequence с номером пакета. Когда количество последовательностей STC\_sequence равно 1, поскольку поле SPN\_STC\_start соответствует началу файла АВ потока клипа, значение поля SPN\_STC\_start равно "0". Поле presentation\_start\_time и поле presentation\_end\_time описывает разрешенный диапазон АВ потока клипа. Ссылка на диапазон, описанный в поле presentation\_start\_time и в поле presentation\_cnd\_time, может быть выполнена из элемента воспроизведения.

На фиг. 52 показан синтаксис, который описывает пример структуры блока ProgramInfo(). Структура синтаксиса блока ProgramInfo() для носителя записи может применяться для носителя, предназначенного только для воспроизведения. Поскольку в блоке ProgramInfo() не используется новая структура, подробное его описание не приведено. В блоке ProgramInfo(), в качестве ограничений может быть описана величина "1" в поле num\_of\_program\_sequences, и значение "1" может быть описано в поле num\_of\_groups.

На фиг. 53 показан синтаксис, который описывает пример структуры блока StreamCodingInfo(). Аналогично блоку ProgramInfo(), блок StreamCodingInfo() имеет структуру синтаксиса, аналогичную структуре записываемой среды. В отношении видеоданных, блок StreamCodingInfo() описывает информацию атрибута формата видеоданных, частоты кадров и соотношения размеров. Что касается звуковых данных, блок StreamCodingInfo() описывает информацию атрибута частоты выборки и т. д. Когда структура синтаксиса носителя, предназначенного для записи, применяется к носителю, предназначенному только для воспроизведения, как показано на фиг. 53, к блоку StreamCodingInfo() необходимо добавить поле language\_code, которое описывает структуру подзаголовков и потоков звуковых данных. Эта информация эффективна, когда наиболее подходящий язык для звука и подзаголовков выбирают в соответствии с установками устройства воспроизведения.

На фиг. 54 показан синтаксис, который описывает пример структуры блока СРІ(). Обычно в закодированном потоке, таком, как поток MPEG, в котором выполняется сжатие между кадрами, поток может быть декодирован в ограниченных положениях, например, в начале ГИЗ (GOP - группа изображения). ИХТ (СРІ информация характерной точки) представляет собой базу данных, которая содержит набор данных о начальных положениях, в которых данные могут быть декодированы. База данных обеспечивает корреляцию времени воспроизведения и адресов файла. В таблице ИХТ содержатся данные, которые описывают начальные положения, в которых могут быть декодированы

данные.

Когда данные воспроизводят с любого момента времени, благодаря ссылке на ИХТ, используемой в качестве такой базы данных со временем воспроизведения, может  
5 быть получен адрес положения воспроизведения в файле. Поскольку этот адрес представляет собой начало, с которого можно декодировать данные, устройство воспроизведения может считывать данные с него и быстро отображать изображение.

10 Начальное положение, в котором данные могут быть декодированы (в данном примере, начальное положение ГИЗ), описано в ИХТ и называется входом в ТВ (EP точка входа).

15 Поле `CPI_type` описывает тип ИХТ. Поле `CPI_type` описывает значение, показанное на фиг. 55. В соответствии с настоящим изобретением, тип поля `CPI_type` описывает ИХТ для носителя, предназначенного только для воспроизведения. В действительности, значение поля `CPI_type` равно "8", что определяет структуру входа ТВ для HDMV  
20 (`EP_map_type_for_HDMV`).

На фиг. 56 показан синтаксис, который описывает пример структуры данных блока `EP_map` входа ТВ для носителя, предназначенного только для воспроизведения, другими  
25 словами, блока `EP_map_for_HDMV()` предыдущего поля `CPI_type`.

Карта `EP_map` представляет собой таблицу, в которой скоррелированы моменты времени воспроизведения и адреса файлов для начальных положений ГИЗ. В примере, показанном на фиг. 56, в качестве таблицы создана база данных, в которой  
30 скоррелированы ШВП (PTS - штампы времени представления) MPEG и НПИ (SPN - номер пакета источника) для начальных положений ГИЗ. НПИ представляет номер пакета источника, который соответствует адресу начального байта файла.

35 Структура карты `EP_map` для носителя, предназначенного для записи, практически совпадает со структурой карты `EP_map` для носителя, предназначенного только для воспроизведения. В соответствии с данным вариантом выполнения, для уменьшения количества данных и ускорения поиска, выполняется грубый поиск и точный поиск по  
40 отдельным значениям. Таким образом, внутренняя структура карты `EP_map` разделена на два цикла, соответствующие грубому поиску и точному поиску. Следовательно, карта `EP_map` для диска, предназначенного только для воспроизведения, является более  
45 сложной, чем простая таблица, "которая коррелирует ШВП первых *i* изображений ГИЗ и адреса файла".

Поле `EP_fine_table_start_address` описывает положение таблицы, используемой для  
50 точного поиска. Следующий цикл `for` описывает таблицы для грубого поиска. Цикл `for` описывает поля `PTS_EP_coarse` и `SPN_EP_coarse`. Поле `ref_to_EP_fine_id` описывает номер

таблицы для точного поиска, ссылка на который делается по результатам грубого поиска. Поля PTS\_EP\_coarse и SPN\_EP\_coarse описывают биты старшего порядка PTS и SPN.

5 После цикла for следует слово заполнения. После слова заполнения следует цикл for, в котором описаны поля PTS\_EP\_fine и SPN\_EP\_fine. Кроме того, в цикле for описан  
10 флаг is\_angle\_change\_point и поле I\_end\_position\_offset. Флаг is\_angle\_change\_point описывает, соответствует ли каждая точка ТВ точке с возможностью смены угла обзора, когда АВ поток клипа имеет структуру множества углов обзора (второй флаг).

Далее будет описан флаг is\_angle\_change\_point. Прежде всего, для простоты понимания будет описано расположение АВ потока клипа на диске.

15 АВ поток клипа состоит из последовательности (блока данных), в которых логические адреса на диске 10 расположены последовательно. Ниже блок данных, как АВ поток клипа, логические адреса которого последовательно расположены на диске, называются протяженностью. На фиг. 57 схематично показана компоновка  
20 протяженностей на диске. В примере, показанном на фиг. 57, один АВ поток клипа состоит из последовательности протяженностей 601А, 601В и 601С. Длины протяженностей и их компоновка соответствующим образом контролируются при  
25 производстве диска.

Далее, протяженности 601А, 601В и 601С структурированы так, что они удовлетворяют правилу минимального размера. Когда АВ поток клипа был записан на  
30 диск 600, была обеспечена последовательная подача воспроизводимых данных.

30 Когда один файл состоит из множества протяженностей 601А, 601В и 601С, как показано на фиг. 57, операция считывания повторяется для последовательного считывания, например, одной протяженности 601А, и затем следующей протяженности  
35 601В. Когда считывают одну протяженность 601А и затем считывают следующую протяженность 601В, необходимо изменять количество оборотов диска 600 и перемещать блок оптической головки, который составляет часть считывания устройства  
40 воспроизведения для диска 600.

Максимальное время доступа при перемещении блока оптической головки по диску 600 составляет приблизительно 0,8 секунды. Пока блок оптической головки перемещается, с привода не могут быть получены данные, которые составляют участок  
45 считывания диска. С другой стороны, поскольку видеоданные и звуковые данные декодируют и воспроизводят последовательно, необходимо использовать механизм компенсации времени доступа.

50 Когда обеспечивают последовательный доступ к одной протяженности 601А и следующей протяженности 601В, между ними происходит переход. Когда длина в байтах

предыдущей протяженности 601А велика, перед выполнением перехода достаточное количество данных может быть считано из протяженности 601А и записано в буфер. Таким образом, при выполнении перехода, поскольку данные, записанные в буфер, поступают в декодер, обеспечивается возможность последовательного декодирования данных.

Другими словами, если размер протяженности 601А, из которой выполняют переход, достаточно велик, при переходе к следующей протяженности 601В предполагается последовательная передача потока. При этом при записи достаточного количества данных в буфер, нижний предел размера протяженности должен быть ограничен. Минимальный размер протяженности определяется как значение, соответствующее скорости считывания диска 600 и времени доступа к приводу. Условие последовательной передачи данных может быть определено в зависимости от того, действительно ли данные были скомпонованы в соответствии с правилом минимального размера протяженности.

На фиг. 58 показан пример, в котором АВ поток клипа был записан на диск по фрагментам, в виде, например протяженностей 601D и 601Е. Для обеспечения возможности считывания записанного АВ потока клипа с заданной скоростью следования битов с носителя записи, показанного на фиг. 58, размер одной протяженности должен удовлетворять условию, определяемому следующей формулой (1).

$$S \times 8 / (S \times 8 / Rud + Ts) \geq Rmax... \quad (1)$$

В формуле (1) "S" представляет размер протяженности в байтах, и "Ts" представляет время доступа с полным ходом, а именно максимальное время доступа, в секундах от одной протяженности 601D до следующей протяженности 601Е. "Rud" представляет скорость битов, при которой данные считывают с диска. "Rmax" представляет скорость следования битов АВ потока клипа. "Rud" и "Rmax" выражены в единицах битов/секунду. Коэффициент "8", на который умножают размер S протяженности, используется для преобразования байт/бит.

Другими словами, когда данные АВ потока клипа последовательно располагаются так, что размер каждой из протяженностей 601D и 601Е становится равным S байт или больше, последовательность протяженностей 601D и 601Е может быть воспроизведена без стыков.

Формула (1) может быть модифицирована как формула (2).

$$S \times 8 \geq Rmax \cdot Rud \cdot Ts / (Rud - Rmax) ... \quad (2)$$

Когда рассчитывают реальное значение в соответствии с формулой (2), если скорость потока битов Rmax АВ потока клипа составляет 28 (бит/секунду), и если

предположить что  $R_{ud} = 35$  Мбит/с (мегабит в секунду) и  $T_s = 0,8$  секунды, удовлетворяется соотношение  $S \geq 14$  Мбайт (мегабайт). Другими словами, в этом примере, минимальный размер  $S$  протяженности равен 14 Мбайт.

5 Аналогично приведенному выше, при последовательном считывании одного файла может быть определено условие последовательной подачи данных. Это условие может быть применено к функции переключения угла обзора без стыков для переключения углов  
10 обзора без стыков, которые считывают при выполнении функции множества углов.

Ниже, со ссылкой на фиг. 59А и фиг. 59В, будет описана функция множества углов. Как, например, показано на фиг. 59А, функция множества углов обеспечивает область для множества частей воспроизведения (эта область называется блоком  
15 множества углов). Когда пользователь воспроизводит эту область, он или она может свободно переключать один угол на другой угол обзора. Когда один и тот же объект снят с помощью множества камер под различными углами обзора, может быть сформирован  
20 такой блок множества углов.

Изображения, которые составляют блок множества углов, не ограничиваются одной сценой. В качестве альтернативы, блок множества углов может состоять из  
25 множества изображений, которые избирательно отображают в один период времени на основе временной оси воспроизведения.

Блок множества углов состоит из одного элемента воспроизведения. Другими словами, один элемент воспроизведения содержит изображения множества углов обзора.  
30 В примере, показанном на фиг. 59А, в блоке множества углов может быть выбран один из трех углов (угол (0), угол (1) и угол (2)).

В качестве примера, показанного на фиг. 59В, для сокращения времени доступа, в течение которого один угол переключают на другой угол, потоки, которые составляют  
35 блок множества углов, записывают на диск со специальной компоновкой. Этот блок, записанный на диск со специальной компоновкой, называется блоком чередования. Один последовательный участок называется модулем чередования. Модуль чередования  
40 записывают в последовательных адресах на диске. Модуль чередования удовлетворяет условию минимального размера протяженности. В блоке чередования модули чередования расположены на временной оси в соответствии с углами.

45 Когда выполняют переключение с одного угла на другой, в каждом блоке чередования, поскольку переход выполняют после считывания данных с минимальным размером протяженности, вышеуказанное условие последовательной передачи данных удовлетворяется. Когда начало ГИЗ соответствует началу модуля чередования, поскольку  
50 декодирование MPEG может быть выполнено от начала модуля чередования, может быть



выполнено переключение без стыков с одного угла обзора на другой угол.

Ниже будет описано положение, в котором может быть выполнено переключение без стыков с одного на другой угол. Очевидно, что один угол нельзя свободно переключать на другой угол в любом байтовом положении, как из-за характеристики потока MPEG, который сжат между кадрами, так и из-за условия последовательной подачи данных с диска. В качестве простейшего способа, один угол переключают на другой угол в каждом модуле чередования, который удовлетворяет минимальному размеру протяженности. Кроме того, когда начало модуля чередования соответствует началу ГИЗ, а именно входу ТВ, обеспечивается возможность последовательного декодирования АВ потока MPEG.

Когда количество точек переключения углов обзора увеличивается, и количество блоков чередования увеличивается в результате чередования, один файл получается фрагментированным. В результате этого увеличивается количество информации управления файловой системой. Например, в файловой системе диска типа "голубой луч", когда файл разделяют на множество протяженностей, увеличивается количество областей диска. В результате этого база данных файловой системы становится большой. В результате, верхнее ограничение по возможному количеству воспроизводимых файлов уменьшается, что является нежелательным.

Таким образом, в соответствии с вариантом выполнения настоящего изобретения, как показано на фиг. 60А и фиг. 60В, размер одного модуля чередования увеличивается так, что количество модулей чередования одного блока чередования уменьшается, и множество точек, в которых возможно переключение углов обзора, описано в одном блоке чередования. Поскольку точка, в которой возможно переключение угла обзора, представляет собой начальную точку декодирования, точка переключения угла обзора соответствует одной из точек ТВ.

Вышеприведенный флаг `is_angle_change_point` представляет, действительно ли вход ТВ, который указывает на начало ГИЗ, соответствует точке, в которой возможно переключение угла. Флаг `is_angle_change_point` имеет значение, показанное на фиг. 61. Когда значение флага `is_angle_change_point` равно "1", он представляет, что вход ТВ представляет собой точку, в которой возможно переключение угла обзора. Когда значение флага `is_angle_change_point` равно "0", он представляет, что вход ТВ не является точкой, в которой возможно переключение углов.

Является ли вход ТВ точкой, в которой возможно переключение угла без стыков, в принципе, зависит от того, действительно или нет вход ТВ и предыдущая точка, в которой возможно переключение углов, расположены друг от друга на расстоянии минимального

размера протяженности или больше. Другими словами, расстояние между входами ТВ, для которых один угол может быть переключен на другой угол, в которых значение флага is\_angle\_change\_point равно "1", должны находиться друг от друга на расстоянии минимального размера протяженности или больше.

Ниже, со ссылкой на фиг. 60А и фиг. 60В, описана операция переключения угла без стыков с использованием флага is\_angle\_change\_point. На фиг. 60А и фиг. 60В точки, обозначенные направленными вверх стрелками, представляют точки, в которых возможно переключение углов. В качестве примера, показанного на фиг. 60В, точки, обозначенные направленными вверх стрелками, представляют собой входы ТВ, которые представляют собой точки, в которых возможно переключение углов. В каждом из этих входов ТВ значение флага is\_angle\_change\_point равно "1".

Как показано на фиг. 60А, предполагается, что, при воспроизведении одного угла (0) пользователь переключает угол (0) на угол (1). Затем устройство воспроизведения последовательно считывает поток угла (0) до ближайшего входа ТВ, значение флага is\_angle\_change\_point которого равно "1". После этого устройство воспроизведения выполняет поиск файла информации клипа для входа ТВ угла (1) синхронного с полученным входом ТВ, получает положение байта полученного входа ТВ, и начинает считывать поток из полученного положения байта. При выполнении описанной выше операции обеспечивается возможность переключения углов без стыков.

На пути воспроизведения на диске, в положении, в котором значение флага is\_angle\_change\_point равно "1", выполняется переход между углами.

Ниже будет описан, случай, в котором один угол переключают на другой угол не без стыков. Переключение угла не без стыков представляет собой переключение угла обзора, при котором в изображении может возникать разрыв. При переключении угла обзора без стыков, если определено переключение угла, как описано выше, поток воспроизводят до точки, в которой возможно переключение угла. В точке, в которой возможно переключение угла, выполняют переключение угла. В отличие от этого, при переключении угла не без стыка, когда обозначается переключение угла, углы можно быстро переключать без выполнения условия последовательной подачи данных и начала ГИЗ. При этом переключение угла не без стыков имеет преимущество, состоящее в коротком времени переключения.

На фиг. 62А и на фиг. 62В показан пример блока множества углов, в которых не обеспечивается переключение без стыков. При переключении угла не без стыка, поскольку в изображении может возникнуть разрыв, отсутствует необходимость всегда переходить на другой угол обзора синхронно с текущим углом. Например, при

воспроизведении ГИЗ одного угла, переход может быть выполнен от этого угла в начало ближайшего ГИЗ другого угла, при этом этот ГИЗ следует раньше, чем ГИЗ исходного угла. Как показано на фиг. 62А, когда переход выполняют с одного угла (0) на другой угол (1), выполняют поиск начала ГИЗ угла (1). Например, угол (1) воспроизводят после того, как он немного возвращается назад. На фиг. 62В показан путь воспроизведения на диске для этого случая.

С другой стороны, блок множества углов без возможности переключения, без стыков, должен быть соединен без стыков на его выходе. Другими словами, когда переключение углов не выполняется вообще, как при обычном воспроизведении, для улучшения качества воспроизводимого изображения очень важно не создавать разрывы на входе и выходе блока множества углов.

В соответствии с вариантом выполнения настоящего изобретения, при установке флага `is_angle_change_point`, блок множества углов не без стыков соединяется без стыков с основной частью.

Соединение без стыков между основной частью и блоком углов может быть выполнено относительно просто. Другими словами, когда конец основной части (конец элемента воспроизведения следует непосредственно после входа блока множества углов не без стыков) удовлетворяет условию минимального размера протяженности, основная часть может быть соединена без стыков для любого угла.

С другой стороны, рядом с выходом блока углов в основную часть, когда углы переключают по пути, который показан на фиг. 63А, данные после последней точки переключения до конца угла не удовлетворяют минимальному размеру протяженности. При этом когда переход выполняют на выходе блока множества углов, условие последовательной подачи данных не удовлетворяется. При этом получается пауза изображения. В качестве альтернативы, происходит разрыв, и, например, экран становится черным. На фиг. 63В показан путь воспроизведения диска для этого случая.

В примере, показанном на фиг. 63А, данные от положения, в котором угол (2) переключают на угол (1) до выхода блока множества углов, не удовлетворяют минимальному размеру протяженности. При этом, когда угол (1) перемещается в основную часть, поскольку величина данных, считываемых для угла (1) является недостаточной, происходит разрыв воспроизводимого изображения.

В соответствии с вариантом выполнения настоящего изобретения, для решения проблемы, которая происходит при безстыковом переключении углов, для безстыкового блока множества углов используют флаг `is_angle_change_point`. При воспроизведении безстыкового блока множества углов, при переключении углов,

происходит разрыв в изображении. Таким образом, значение флага `is_angle_change_point` всегда представляет собой "1", как показано на фиг. 63B. Другими словами, переход может быть выполнен из любого положения воспроизводимого угла. Кроме того, воспроизведение может быть выполнено из любого положения перехода.

Ниже будет приведено определение флага `is_angle_change_point` для клипа в небезстыковом блоке множества углов. Как описано выше, в области, в которой значение флага `is_angle_change_point` равно "0", при воспроизведении изображения, переключение угла на другой угол обзора (переход на другой угол) и возврат к положению воспроизведения с исходного угла (возврат к исходному углу) запрещены. В отличие от этого, в области, в которой значение флага `is_angle_change_point` равно "1", можно выполнять "переход на другой угол" и "возврат к исходному углу". В непосредственной близости к выходу каждого угла, в заданном количестве положений, величина флага `is_angle_change_point` установлена равной "0". В области, в которой величина флага `is_angle_change_point` установлена в "0", "переход на другой угол" и "возврат в исходный угол" запрещены. Размер области, в которой значение флага `is_angle_change_point` установлено равным "0", равен или больше, чем минимальный размер протяженности.

Поскольку флаг `is_angle_change_point` установлен таким образом, непосредственно перед выходом, из небезстыкового блока множества углов, может быть запрещен "переход на другой угол" и "возврат на исходный угол".

Например, как показано на фиг. 64B и фиг. 65B, величина флага `is_angle_change_point` установлена равной "0" в шести входах ТВ на оконечной стороне каждого модуля чередования, в то время как значение флага `is_angle_change_point` установлено равным "1" в других входах ТВ каждого модуля чередования.

Ниже будут описаны способы воспроизведения небезстыкового блока множества углов в соответствии со значением флага `is_angle_change_point`, который был установлен так, как описано выше. В первом способе, в области, в которой значение флага `is_angle_change_point` было установлено равным "0", переключение угла запрещено.

Ниже со ссылкой на фиг. 64A и фиг. 64B будет описан первый способ. Как описано выше, для шести входов ТВ на стороне окончания каждого модуля чередования, значение флага `is_angle_change_point` установлено равным "0" (см. фиг. 64B). Как показано на фиг. 64A, когда положение воспроизведения представляет собой область, в которой величина флага `is_angle_change_point` была установлена равной "0", переключение угла запрещено. Даже если от пользователя поступает команда на переключение угла в этой области, устройство воспроизведения будет игнорировать команду. В данном примере устройство воспроизведения не выполняет переключение угла. Устройство воспроизведения выходит

из блока множества углов и входит в элемент воспроизведения основной части. На фиг. 64В показан путь воспроизведения на диске для этого случая.

Во втором способе, хотя переключение угла разрешено в области, в которой значение флага `is_angle_change_point` было установлено равным "0", после выполнения переключения угла выполняется переход в положение, в котором величина флага `is_angle_change_point` была установлена равной "1".

Ниже со ссылкой на фиг. 65А и фиг. 65В, будет описан второй способ. Как описано выше, для шести входов ТВ на стороне окончания каждого модуля чередования значение флага `is_angle_change_point` установлено равным "0" (см. фиг. 65В). Как показано на фиг. 65А, в положении, расстояние от которого до выхода блока множества углов меньше, чем минимальный размер протяженности, а именно в области, в которой значение флага `is_angle_change_point` было установлено равным "0", если определено переключение угла, положение воспроизведения возвращается к области, в которой значение флага `is_angle_change_point` было установлено равным "1", и воспроизведение начинается из этой области. На фиг. 65В показан путь воспроизведения на диске для этого случая.

В любом из первого и второго способов, когда выполняют переход на выходе из блока множества углов, после считывания данных с минимальным размером протяженности или больше, воспроизводится следующий элемент воспроизведения. При этом предотвращается разрыв (небезстыковость) на выходе из блока углов.

На фиг. 64А и фиг. 64В блок чередования состоит из небезстыкового блока множества углов. Однако, как показано на фиг. 65А и фиг. 65В, другой блок, помимо блока чередования, может содержать небезстыковый блок множества углов.

На фиг. 66А, фиг. 66В и фиг. 66С показаны схемы функционального блока, представляющие пример структуры декодера 100 устройства воспроизведения в соответствии с вариантом выполнения настоящего изобретения. Декодер 100 устройства воспроизведения интерпретирует данные, воспроизводимые с диска, загруженного в устройство привода (не показано), выводит АВ поток и обеспечивает для пользователя возможность интерактивной работы с выходным АВ потоком.

Управление всеми операциями декодера 100 устройства воспроизведения выполняют с помощью ЦПУ (не показано). Потоки и данные отдельных частей декодера устройства 100 воспроизведения отслеживаются и управляются с помощью ЦПУ.

Когда диск загружен в устройство привода (не показано), как описано выше, воспроизводится файл "scenario.hdmv" и файл "entrylist.data". В соответствии с описаниями файла "scenario.hdmv" и файла "entrylist.data", другие необходимые файлы считывают с диска, и, таким образом, воспроизводят содержание, записанное на диск.

Например, в соответствии с описаниями файла "scenario.hdmv" и файла "entrylist.data", с диска считывают данные подвижного изображения, отображаемые на плане 10 подвижного изображения, данные изображения, отображаемые на плане 11 подзаголовка, и плане 12 графического изображения, файл списка воспроизведения, и т. д.

В следующем описании, среди этих данных, которые считывают с диска, такие потоки, как данные подвижного изображения, подизображения (данные подзаголовка), и данные звука, которые должны непрерывно обрабатываться, называются потоками, поступающими в режиме реального времени. В отличие от этого, данные, не поступающие в режиме реального времени, такие как файлы сценария и файлы списка воспроизведения, для которых не требуется последовательная обработка, называются сохраненными объектами. Сохраненные объекты записаны в запоминающем устройстве или подобном устройстве, в котором их расширяют. Сохраненные объекты записаны в запоминающем устройстве или подобном устройстве, их расширяют и обрабатывают по мере необходимости.

Декодер 100 устройства воспроизведения имеет две системы входных каналов, которые представляют собой канал (1) и канал (2). Сохраненный объект поступает на вход 101 входного канала (1). Поток в режиме реального времени поступает на вход 202 входного канала (2). В качестве альтернативы, сохраненный объект может поступать на вход 202. В соответствии с вариантом выполнения, поток в режиме реального времени и часть сохраненного объекта, которая поступает на вход 202, представляет собой MPEG2 ТП.

Когда скорость вращения диска в устройстве привода повышается, например, в два раза, и скорость передачи считываемых данных с диска увеличивается, операции считывания с диска для двух систем каналов (1) и (2) выполняются на основе деления времени.

Прежде всего, будет описана система входного канала (1). Сохраненный объект, который поступает на вход 101, поступает в схему 102 переключения. Когда на вход, в качестве сохраненного объекта, поступает код программы файла "scenario.hdmv", схема 102 переключения выбирает выход 102А. Подаваемый на вход программный код записывают в буфер 104 кода. В качестве другой программы может поступать файл HTML (язык разметки гипертекста), сценарий ECMA (Европейская ассоциация по стандартизации информационных и вычислительных систем) или тому подобное.

Когда данные изображения и данные звука поступают в качестве сохраненного объекта, схема 102 переключения выбирает выход 102В. В результате, входные данные изображения поступают на схему 103 переключения. Когда поток в режиме реального

времени, который поступает на вход 202, не содержат данные изображения, отображаемые на плане 11 подзаголовка или на плане 12 графического изображения, схема 103 переключения выбирает вход 103А. Данные изображения и звуковые данные, которые поступают со схемы 102 переключения, записывают в буфер 105 содержания.

Аналогично, когда данные изображения, отображаемые на плане 11 подзаголовка или плане 12 графических изображений, и звуковые данные в качестве звукового эффекта содержатся в потоке реального времени, который поступает на вход 202, схема 103 переключения выбирает вход 103В. В результате, данные изображения /звуковые данные записывают в буфер 105 содержания. Сохраненные объекты, записанные в буфер 104 кода, и буфер 105 содержания считывают, по мере необходимости и передают в мультимедийный механизм 106.

Данные изображения сохраненного объекта, записанные в буфер 105 содержания, также поступают в декодер 300 А графических изображений и декодер 301 В графических изображений через схемы 107 и 108 переключения, соответственно.

В примере, показанном на фиг. 66А, фиг. 66В и фиг. 66С, декодер 300 А графических изображений декодирует данные изображения в формате ПСГ. Декодер 301 В графических изображений декодирует данные изображения в формате JPEG. В качестве альтернативы, декодер 300 А графических изображений и декодер 301 В графических изображений может декодировать данные изображения в другом формате и/или данные изображения во множестве форматов.

Мультимедийный механизм 106 содержит анализатор 106А синтаксиса XML (расширяемый язык разметки гипертекста), интерпретатор 106В сценария, устройство 106С построения и отображения графического изображения, 106D звука, который воспроизводит звуковой эффект, и механизм 106Е анализа файла сценария. Мультимедийный механизм 106 может быть составлен из независимых аппаратных средств. В качестве альтернативы, мультимедийный механизм 106 может быть выполнен с помощью процесса определенной программы, которая выполняется на указанном выше ЦПУ (не показано).

Анализатор 106А синтаксиса XML выполняет функцию анализа синтаксиса документа XML (расширяемый язык разметки гипертекста). Кроме того, анализатор 106А синтаксиса XML также позволяет анализа синтаксис документа HTML. Документ HTML, анализируемый с помощью анализатора 106А синтаксиса XML, преобразуется в формат, который может выполняться декодером 100 устройства воспроизведения. Интерпретатор 106В сценария анализирует синтаксис сценария ЕСМА (Европейская ассоциация по стандартизации информационных и вычислительных систем) и преобразует его в формат,

который может выполняться декодером 100 устройства воспроизведения. Устройство 106С построения и отображения графических изображений декодирует данные изображения и получает формат, который может быть расширен на план 11 подзаголовка и план 12 графических изображений. Проигрыватель 106D звука воспроизводит звуковые данные, используемые в качестве звука нажатия на кнопку и т. д.

Мультимедийный механизм 106 выполняет процессы для анализатора 106А синтаксиса XML, интерпретатора 106В сценария, и устройства 106С построения и отображения графических изображений с использованием рабочего запоминающего устройства буфера 109. Например, анализатор 106А синтаксиса XML и интерпретатор 106В сценария используют буфер 109а кода буфера 109. Устройство 106С построения и отображения графических изображений использует буфер 109D графических изображений буфера 109. Буфер 109, кроме того, содержит буфер 109В шрифта, в котором записаны данные шрифта, используемые для отображения строки знаков и буфер 109С дерева, в котором записан результат анализа синтаксиса документа HTML, получаемый с помощью анализатора 106А синтаксиса XML в иерархической древовидной структуре. Проигрыватель 106D звука воспроизводит звуковые данные, записанные в буфер 109Е звука.

Мультимедийный механизм 106 считывает файл сценария "scenario.hdmv" и т.д., записанный, например, в буфер 104 кода. Механизм 104Е анализа сценария анализирует сценарий в соответствии с описанием считанного файла. Управление устройством привода или подобным устройством (не показано) выполняют в соответствии с описанием анализируемого сценария. Необходимые файлы для данных подвижного изображения, отображаемого в плане 100 подвижного изображения, данные изображения, отображаемые в плане 11 подзаголовка1 и плане 12 графических изображений, звуковые данные, и т. д. считывают с диска, загруженного в устройство привода. В результате, обеспечивается воспроизведение содержания.

Данные, которые записаны в буфер 104 кода и буфер 105 содержания, могут содержаться в нем до тех пор, пока эти данные не станут ненужными. При этом данные, записанные в буфер 104 кода и буфер 105 содержания, можно многократно считывать в случае необходимости.

Кроме того, мультимедийный механизм 106 считывает сценарий ЕСМА из буфера 104 кода. В случае необходимости мультимедийный механизм 106 считывает другой сценарий ЕСМА и другой документ HTML из буфера 104 кода в соответствии со считанным описанием сценария ЕСМА. Кроме того, мультимедийный механизм 106 считывает данные изображения из буфера 105 содержания.



Кроме того, мультимедийный механизм 106 выполняет процесс демультимплексирования для множества типов входных данных, функцию JavaVM (виртуальная машина Java (зарегистрированный товарный знак)), и т.д. Кроме того, мультимедийный механизм 106 принимает входные команды пользователя из средства управления такого, как дистанционный пульт управления, устройство-указатель или тому подобное (не показано) и выполняет процесс в соответствии с входными командами пользователя. Входные команды пользователя также поступают в декодер 300 А графических изображений, декодер 301 В графических изображений, декодер 118 звука, декодер 120 MPEG видео, и системный декодер 121, который будет описан ниже.

Данные изображения, обработанные устройством 106С построения и отображения графических изображений, поступают в план 302 подизображения и план 303 графических изображений через схемы 130 и 131 переключателей, соответственно. В этом примере предполагается, что входные данные изображения, поступающие в план 302 подизображения и в план 303 графических изображений, представляют собой данные растрового изображения, в которых данные изображения в формате ПСГ, формате JPEG или подобном формате были декодированы с помощью устройства 106С построения и отображения графических изображений. Моментами синхронизации, в которые эти данные изображения поступают в планы 302 и 303, управляют с помощью мультимедийного механизма 106.

План 302 подизображения и план 303 графических изображений соответствуют вышеуказанному плану 11 подзаголовка и плану 12 графических изображений, соответственно. План 135 подвижного изображения соответствует вышеуказанному плану 10 подвижного изображения. Каждый из этих планов 302 подизображения, плана 303 графического изображения и плана 135 подвижного изображения состоит, например, из запоминающего устройства кадра.

Мультимедийный механизм 106 также передает сигнал управления, который обеспечивает выбор одного из планов 135 подвижного изображения, плана 302 подизображения и плана 333 графического изображения или комбинирования их в процессоре 139 представления, который будет описан ниже. Аналогично, мультимедийный механизм 106 передает сигнал управления, с помощью которого обеспечивается управление выходом звукового потока в процессор 141 представления, который будет описан ниже.

Далее будет описана система входного канала (2). Поток в режиме реального времени, который поступает как MPEG2 ТП на вход 202, передают на фильтр 110 ИДП. Фильтр 110 ИДП выделяет личный ИДП (идентификатор пакета) из MPEG2 ТП

транспортного потока и детектирует атрибут потока, содержащийся в транспортном пакете. Фильтр 110 ИДП разделяет входной поток в режиме реального времени по соответствующим системам для каждого транспортного пакета в соответствии с атрибутом потока.

Когда транспортный пакет представляет собой пакет, в котором содержатся данные изображения сохраненного объекта, транспортный пакет временно записывают в буфер ТВп 111А. Транспортный пакет считывают в заранее определенные моменты синхронизации и подают на вход схемы 103 переключателя через выбранный вход 103В. После этого транспортный пакет записывают в буфер 105 содержания через схему 103 переключателя.

Когда фильтр 110 ИДП определяет, что транспортный пакет содержит данные формата ПСГ в соответствии с ИДП, транспортный пакет временно записывают в буфер ТВп 111В и буфер Вп 112В. Транспортный пакет считывают в заранее определенные моменты синхронизации и подают на вход схемы 107 переключателя через выбранный вход 107В. Транспортный пакет подают в декодер 300 А графических изображений через схему 107 переключателя.

Декодер 300 А графических изображений удаляет информацию заголовка из переданного на него транспортного пакета, декодирует данные ПСГ, содержащиеся в транспортном пакете и получает данные изображения для подзаголовков или графических изображений. Когда данные изображения используют в качестве подзаголовков, данные изображения подают на вход 130В схемы 130 переключения, и расширяют в плане 302 подизображения через схему 130 переключения в заранее определенные моменты синхронизации. Аналогично, когда используют данные изображения в виде кнопок, поскольку их отображают в плане графических изображений, данные изображения подают на вход 131С схемы 130 переключения и расширяют в плане 303 графического изображения через схему 131 переключения.

Когда фильтр 110 ИДП определяет, что транспортный пакет содержит данные JPEG в соответствии с ИДП, транспортный пакет временно записывают в буфер ТВп 111С и буфер Вп 112С. Транспортный пакет считывают в заранее определенный момент синхронизации, подают в схему 108 переключения через выбранный вход 108В и передают в декодер 301 В графических изображений через схему 108 переключателя.

Декодер 301 В графических изображений удаляет информацию заголовка из переданного транспортного пакета, декодирует данные JPEG, содержащиеся в транспортном пакете и получает данные растрового изображения. Когда данные изображения используют в качестве подзаголовков, данные изображения подают на вход

130С схемы 130 переключения в заранее определенные моменты времени, и расширяют до плана 302 подизображения через схему 130 переключения. Аналогично, когда данные изображения используются в качестве кнопок, данные изображения поступают на вход 131В схемы 131 переключения в определенные моменты синхронизации и расширяются в плане 302 графических изображений через схему 131 переключения.

Когда фильтр 110 ИДП определяет, что транспортный пакет содержит звуковые данные в соответствии с ИДП, транспортный пакет временно записывают в буфер ТВп 111D и Вп 112D. Транспортный пакет считывают в заданные моменты синхронизации и передают в звуковой декодер 118. Данные звука, содержащиеся в транспортном пакете, кодируют со сжатием в соответствии с системой, основанной, например, на системе обработки Dolby Digital.

Декодер 118 звука содержит, например, линейный звуковой декодер 119 ИКМ (PCM - с импульсно-кодовой модуляцией), который удаляет информацию заголовка из входного транспортного потока, декодирует звуковые данные, закодированные со сжатием, содержащиеся в транспортном пакете, и, наконец, получает линейные звуковые данные ИКМ. Когда транспортный пакет содержит линейные звуковые данные ИКМ, которые не были закодированы со сжатием, эти данные подают непосредственно на линейный звуковой декодер 119 ИКМ. С выхода линейного звукового декодера 119 ИКМ выводятся данные, без выполнения какой-либо обработки.

Линейные звуковые данные ИКМ, которые поступают с выхода звукового декодера 118, подают в процессор 141 представления для звука. В процессоре 141 представления заданный звуковой эффект и т.д. добавляют к линейным звуковым данным ИКМ под управлением мультимедийного механизма 106, и затем их передают на выход 142.

Когда фильтр 110 ИДП определяет, что транспортный пакет содержит данные подвижного изображения в соответствии с ИДП, транспортный пакет временно записывают в буфер ТВп 111E, буфер МВп 113 и буфер ЕВп 114, считывают в заранее определенные моменты синхронизации и передают в видеodeкодер 120 MPEG. Данные подвижного изображения, содержащиеся в транспортном пакете, закодированы со сжатием в соответствии с системой MPEG2.

Видеodeкодер 120 MPEG удаляет информацию заголовка из переданного транспортного пакета, декодирует данные подвижного изображения, которые были закодированы со сжатием в соответствии с системой MPEG2, и получает данные подвижного изображения в полосе модулирующего сигнала.

Данные подвижного изображения, которые поступают с выхода видеodeкодера 120 MPEG, поступают на вход 124А схемы 124 переключателя. Кроме того, данные

подвижного изображения подают на вход 124В схемы 124 переключателя через буфер 123. В схеме 124 переключателя входы 124А и 124В выбирают в заранее определенный момент синхронизации. Выходные данные подвижного изображения расширяют в плане 134 подвижного изображения.

Когда фильтр 110 ИДП определяет, что транспортный пакет содержит системную информацию в соответствии с ИДП, транспортный пакет передают в системный декодер 121 через буферы ТВп 111F и Bsys 115. Системный декодер 121 удаляет информацию заголовка из переданного транспортного пакета и выделяет из него системную информацию. Системную информацию подают, например, в ЦПУ (не показано).

Данные изображения на плане 302 подизображения подают в палитру 304, которая соответствует описанной выше таблице 22 палитры, показанной на фиг. 17. Палитра содержит 256 цветов. Ссылку на палитру обеспечивают с помощью индекса. На выход поступают данные RGB. Кроме того, выделяют данные  $\alpha_1$  степени прозрачности. Данные RGB преобразуют в данные YCbCr с помощью схемы 306 преобразования RGB/YCbCr, которая соответствует описанной выше схеме 29 преобразования RGB/YCbCr, показанной на фиг. 17. В процессор 139 представления, кроме данных  $\alpha_1$  степени прозрачности, подают данные YCbCr.

Данные изображения в плане 303 графического изображения подают в палитру 305, которая соответствует описанной выше таблице 26А палитры, показанной на фиг. 17. В результате, на выход поступают данные RGB и данные  $\alpha_2$  степени прозрачности. Данные RGB подают в схему 307 преобразования RGB/YCbCr, которая соответствует схеме 26В преобразования RGB/YCbCr, показанной на фиг. 17. В результате, система цвета преобразуется из RGB (4 : 4 : 4) в YCbCr (4 : 4 : 4). Кроме данных  $\alpha_2$  степени прозрачности, данные YCbCr, поступают с выхода схемы 307 преобразования RGB/YCbCr в процессор 139 представления.

Выход плана 135 подвижного изображения поступает в процессор 139 представления через преобразователь 138 вверх/вниз. Преобразователь 138 вверх/вниз представляет собой схему, которая преобразует разрешающую способность изображения. Преобразователь 138 вверх/вниз преобразует, например, изображение ВР (HD - с высокой разрешающей способностью), которое имеет высокое разрешение в изображение СР (SD – со стандартным разрешением), которое имеет стандартное разрешение.

Процессор 139 представления выполняет процесс альфа-смешения, с использованием степени  $\alpha_1$  прозрачности данных изображения плана 11 подзаголовка (план 302 подзаголовка), и степени  $\alpha_2$  прозрачности плана 12 графических изображений (плана 303 графических изображений), описанных со ссылкой на фиг. 17.

Процессор 139 представления комбинирует данные изображения плана 135 подвижного изображения и данные изображения плана 302 подизображения в соответствии со степенью  $\alpha_1$  прозрачности, которая была установлена для данных изображения плана 502 подизображения. Кроме того, процессор 139 представления комбинирует данные изображения, из которых были скомбинированы данные изображения плана 135 подвижного изображения и данные изображения плана 302 подзаголовка, и были установлены данные изображения плана 303 графического изображения, в соответствии со степенью  $\alpha_2$  прозрачности, для данных изображения плана 303 графического изображения. Данные изображения, в которых были скомбинированы данные изображения плана 303 графического изображения, данные изображения (данные подзаголовка) плана 302 подизображения и данные изображения плана 135 подвижного изображения, получают на выходе 140.

Процессор 139 представления может выполнять процесс эффекта для данных изображения, поступающих в режиме реального времени.

В такой структуре, когда описанное содержание, данные которого структурированы в соответствии с вариантом выполнения настоящего изобретения, загружают в устройство привода (не показано), считывается файл "scenario.hdmv" и т. д. В соответствии с описанием файла "scenario.hdmv", воспроизводят содержание, записанное на диск. Когда от пользователя через средство управления (не показано) поступает команда, управление каждой частью устройства привода и декодера 100 устройства воспроизведения выполняют с помощью мультимедийного механизма 106. Данные воспроизводят в соответствии с этой командой.

Когда со средства управления (не показано) поступает команда переключения угла, делается ссылка на требуемые файлы "scenario.hdmv", "entrylist.data", и т.д. под управлением мультимедийного механизма 106. При этом устройством привода (не показано) управляют в соответствии с информацией о точке возможности переключения клипа. В соответствии с приведенным выше описанием, положение считывания на диске изменяют для переключения углов. Кроме того, на средство управления может поступать команда отмены паузы для неподвижного изображения.

В приведенном выше описании каждая часть декодера 100 устройства воспроизведения 100 состоит из аппаратных средств. Однако настоящее изобретение не ограничивается таким примером. Например, декодер 100 устройства воспроизведения может быть выполнен в виде процесса программных средств. В этом случае декодер 100 устройства воспроизведения может работать на компьютерном устройстве. Кроме того, декодер 100 устройства воспроизведения может быть выполнен в виде комбинации

аппаратных и программных средств. Например, звуковой декодер 118 и видеodeкодер 120 MPEG могут состоять из аппаратных средств. Остальная часть декодера 100 устройства воспроизведения может состоять из программных средств.

5 Программа, которая обеспечивает возможность выполнения с помощью компьютерного устройства декодера 100 устройства воспроизведения, состоящего только из программных средств или комбинации аппаратных и программного средств, записана на носитель записи, например, CD-ROM (компакт-диск, предназначенный только для считывания) и поставляется вместе с ним. CD-ROM загружают в привод CD-ROM компьютерного устройства. Программу, записанную на CD-ROM, устанавливают на компьютерное устройство. В результате, вышеописанный процесс может быть выполнен на компьютерном устройстве. Поскольку структура компьютерного устройства хорошо известна, описание ее здесь не будет приведено.

Как описано выше, настоящее изобретение обеспечивает возможность изменения интерактивной функции, обеспечивающей воспроизведение в соответствии с входными командами пользователя, которая может быть выполнена благодаря расширению стандарта диска типа "голубой луч" для записи и воспроизведения данных для предварительно записанного диска большой емкости.

25 Кроме того, в соответствии с вариантом выполнения настоящего изобретения, поскольку информация списка воспроизведения, составляющая заголовок и верхнее меню, описана в базе данных, требуемый заголовок может непосредственно воспроизводиться при нажатии на клавишу пульта дистанционного управления или подобного устройства, без необходимости выбора требуемого заголовка на экране. Кроме того, может быть вызван экран верхнего меню.

35 Кроме того, в соответствии с вариантом выполнения настоящего изобретения, поскольку флаг, который представляет, действительно ли можно переключать углы обзора, описан в начале каждого модуля декодирования, обеспечивается возможность функции переключения угла без стыков, при выполнении которой не возникают разрывы в точке переключения угла обзора.

45 В соответствии с вариантом выполнения настоящего изобретения, поскольку описан флаг, который представляет, возможно ли воспроизведение подэлемента воспроизведения не синхронно с основным элементом воспроизведения, обеспечивается возможность многократного воспроизведения ФМС, не синхронно с основным путем.

### Формула изобретения

50 1. Устройство воспроизведения, предназначенное для воспроизведения данных содержания, записанных на носитель записи в форме диска, содержащее:

средство считывания, предназначенное для считывания с носителя записи видеопотока, первого флага, описанного для каждого воспроизводимого блока

видеопотока, количества углов, которые содержит воспроизводимый блок, и информации о положении, которая представляет угловые положения в видеопотоке, причем первый флаг представляет, действительно ли воспроизводимый блок может воспроизводиться под множеством углов,

5 средство воспроизведения, предназначенное для управления средством считывания для считывания видеопотока, который составляет углы, в соответствии с информацией о положении, и выполненное с возможностью изменения положения воспроизведения видеопотока в соответствии с информацией о положении так, что  
10 обеспечивается возможность переключения текущего угла, когда первый флаг представляет, что воспроизводимый блок можно воспроизводить под множеством углов, в котором видеопоток был закодирован в модуле одного или множества кадров и записан на носитель записи,

15 в котором средство считывания выполнено с возможностью считывания второго флага, описанного в модуле кодирования, с носителя записи, причем второй флаг представляет, действительно ли обеспечивается возможность переключения текущего угла в начале модуля кодирования, и в котором средство воспроизведения сконфигурировано с возможностью изменения положения воспроизведения  
20 видеопотока в положении в соответствии со вторым флагом.

2. Устройство воспроизведения по п.1,

в котором второй флаг описан в заданной области на стороне окончания каждого из углов.

3. Устройство воспроизведения по п.2,

25 в котором средство воспроизведения сконфигурировано так, что оно не изменяет положение воспроизведения, когда поступает команда, которая обеспечивает изменение положения воспроизведения в заданной области.

4. Устройство воспроизведения по п.2,

30 в котором средство воспроизведения сконфигурировано для возврата в положение непосредственно перед заданной областью переключаемого угла и воспроизведения угла, когда поступает команда, которая обеспечивает изменение положения воспроизведения в заданной области.

5. Устройство воспроизведения по п.2,

35 в котором размер заданной области основан на максимальном времени доступа средства считывания из первой области во вторую область на носителе записи и разности между скоростью считывания и скоростью воспроизведения видеопотока средством считывания со средства воспроизведения.

40 6. Способ воспроизведения, предназначенный для воспроизведения данных содержания, записанных на носитель записи в форме диска, содержащий следующие этапы:

считывание с носителя записи видеопотока, первого флага, описанного для  
45 каждого воспроизводимого блока видеопотока, количества углов, которые содержит воспроизводимый блок, и информации о положении, которая представляет угловые положения в видеопотоке, причем первый флаг представляет, действительно ли воспроизводимый блок может быть воспроизведен с множеством углов, и управление этапом считывания для считывания видеопотока, который  
50 составляет углы, в соответствии с информацией о положении, при этом изменяют положения воспроизведения видеопотока в соответствии с информацией о положении так, что обеспечивается возможность переключения текущего угла, когда первый флаг представляет, что воспроизводимый блок можно воспроизводить

под множеством углов, в котором видеопоток был закодирован в модуле одного или множества кадров и записан на носитель записи,

в котором дополнительно осуществляют считывание второго флага, описанного в модуле кодирования, с носителя записи, причем второй флаг представляет, действительно ли обеспечивается возможность переключения текущего угла в начале модуля кодирования, и изменяют положение воспроизведения видеопотока в положении в соответствии со вторым флагом.

7. Носитель записи в форме диска, на который была записана программа воспроизведения, которая может быть считана компьютерным устройством, причем программа воспроизведения обеспечивает возможность выполнения с помощью компьютерного устройства способа воспроизведения, предназначенного для воспроизведения данных содержания, записанных на носитель записи, причем способ воспроизведения содержит следующие этапы:

считывание с носителя записи видеопотока, первого флага, описанного для каждого воспроизводимого блока видеопотока, количества углов, которые содержит воспроизводимый блок, и информации о положении, которая представляет угловые положения в видеопотоке, причем первый флаг представляет, действительно ли воспроизводимый блок может быть воспроизведен с множеством углов,

управление этапом считывания для считывания видеопотока, который составляет углы, в соответствии с информацией о положении, при этом изменяют положения воспроизведения видеопотока в соответствии с информацией о положении так, что обеспечивается возможность переключения текущего угла, когда первый флаг представляет, что воспроизводимый блок можно воспроизводить под множеством углов, в котором видеопоток был закодирован в модуле одного или множества кадров и записан на носитель записи,

в котором дополнительно осуществляют считывание второго флага, описанного в модуле кодирования, с носителя записи, причем второй флаг представляет, действительно ли обеспечивается возможность переключения текущего угла в начале модуля кодирования, и изменяют положение воспроизведения видеопотока в положении в соответствии со вторым флагом.

8. Носитель записи в форме диска, на который были записаны данные содержания, в котором видеопоток, первый флаг, описанный для каждого воспроизводимого блока видеопотока, количество углов, которые содержит воспроизводимый блок, и информацию о положении, которая представляет угловые положения в видеопотоке, были записаны на носителе записи, причем первый флаг представляет, действительно ли воспроизводимый блок может быть воспроизведен с множеством углов, и в котором видеопоток, который составляет углы, считывают в соответствии с информацией о положении, при этом положение воспроизведения видеопотока можно изменять в соответствии с информацией о положении, когда первый флаг представляет, что воспроизводимый блок может быть воспроизведен с множеством углов, в котором видеопоток был закодирован в виде модуля из одного или множества кадров и записан на носитель записи, и

в котором второй флаг был записан в модуле кодирования, причем второй флаг представляет, действительно ли текущий угол можно переключить в начале модуля кодирования.

9. Носитель записи по п.8,

в котором второй флаг описан в заданной области на стороне окончания каждого



из углов.

10. Носитель записи по п.9,

в котором, когда поступает команда, которая обеспечивает изменение положения воспроизведения в заданной области, положение воспроизведения не меняется.

5

11. Носитель записи по п.9,

в котором, когда поступает команда, которая обеспечивает изменение положения воспроизведения в заданной области, прослеживают положение, следующее непосредственно перед заданной областью переключаемого угла, и переключаемый

10

угол воспроизводят из прослеженного положения.

12. Носитель записи по п.9,

в котором размер заданной области основан на максимальном времени доступа из первой области во вторую область и на разности между скоростью считывания и скоростью воспроизведения видеопотока.

15

20

25

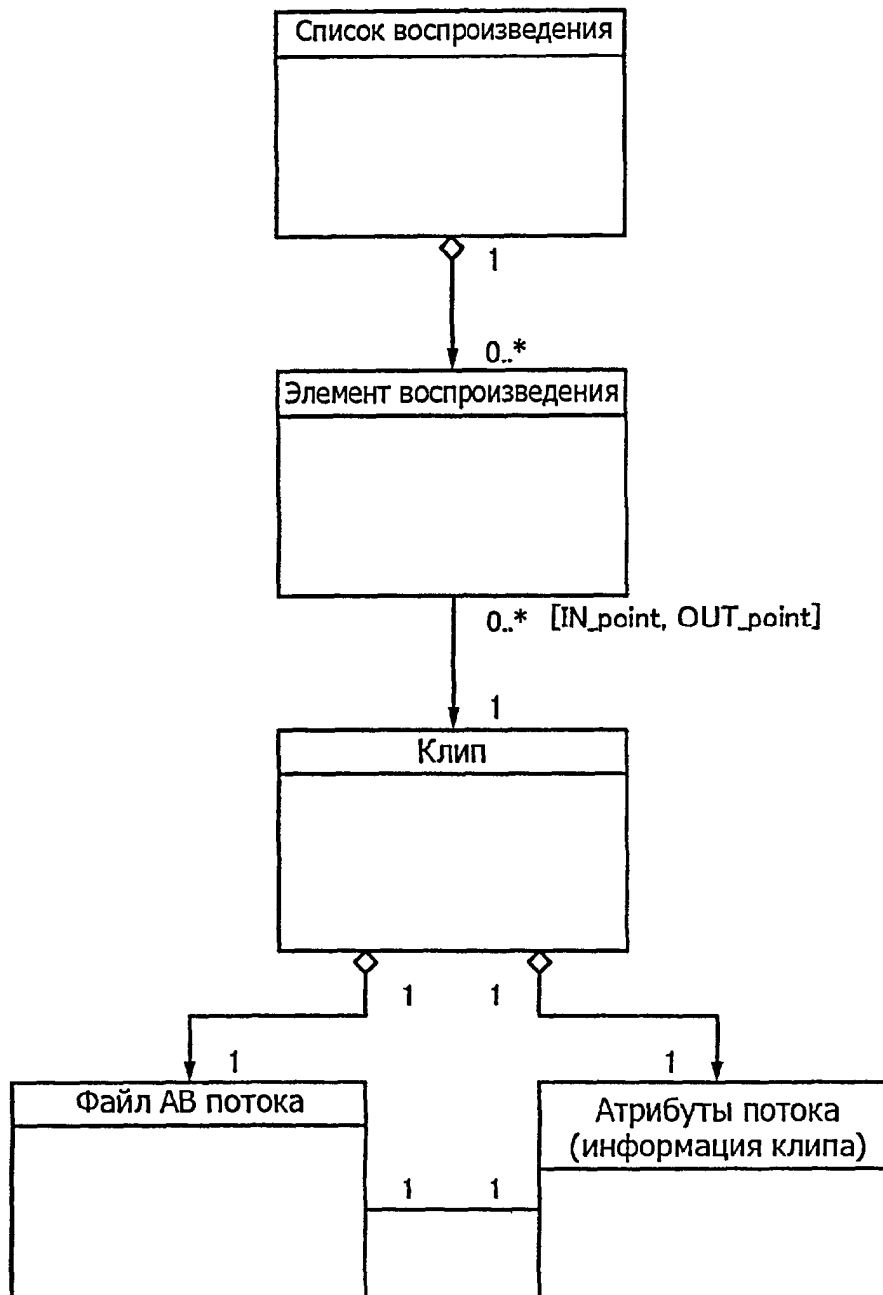
30

35

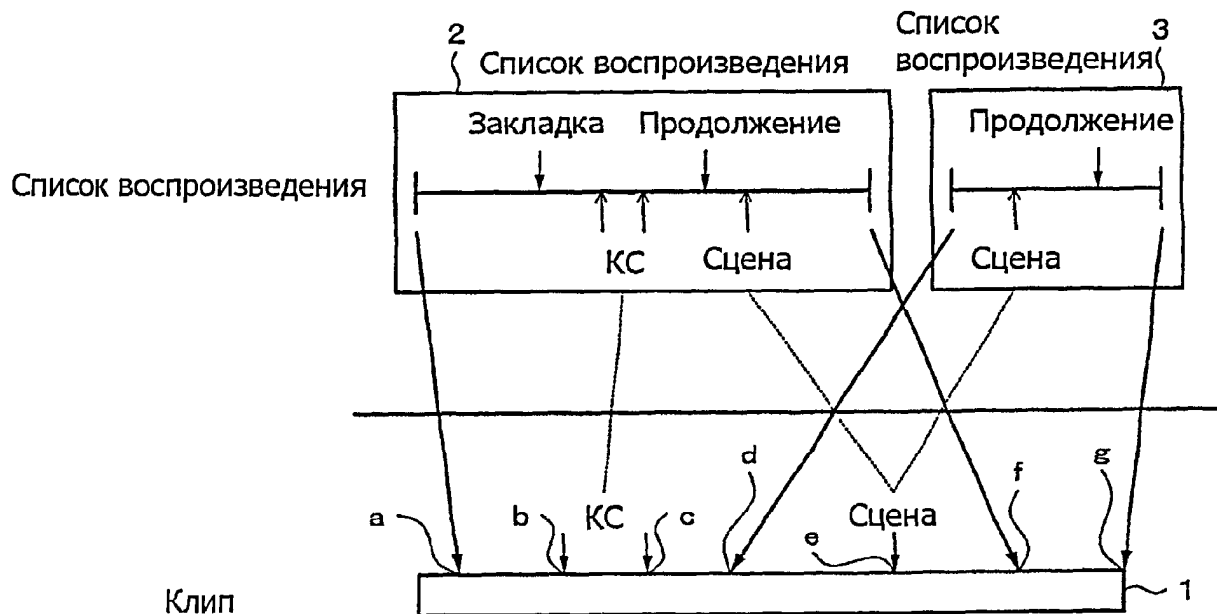
40

45

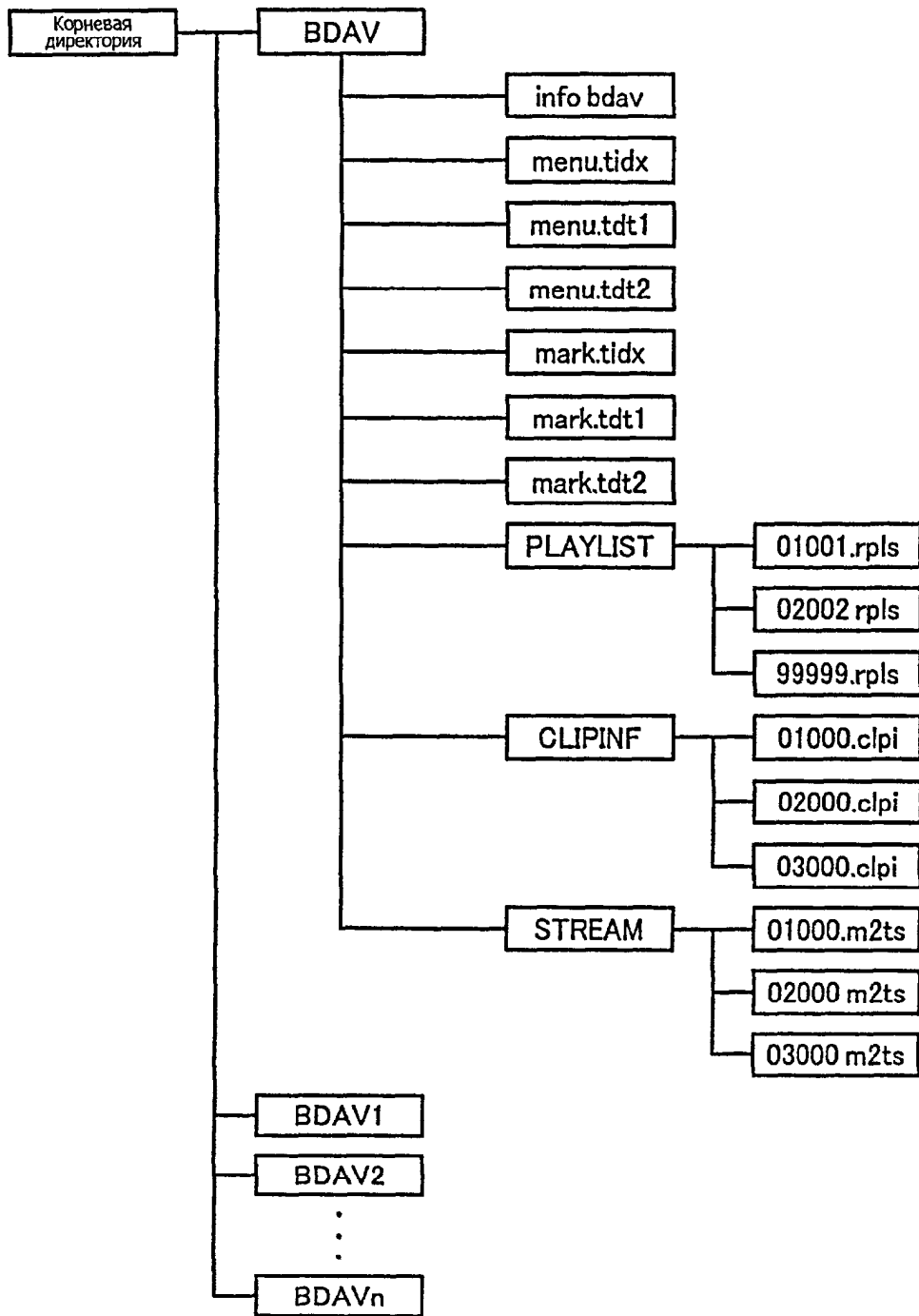
50



ФИГ. 2



ФИГ. 3



ФИГ. 4

СИНТАКСИС	Длина данных (бит)	Мнемоническая схема
info.bdav{		
type_indicator	8*4	bslbf
version_number	8*4	bslbf
TableOfPlayLists_start_address	32	unimsbf
MakersPrivateData_start_address	32	unimsbf
reserved_for_future_use	192	bslbf
UIAppInfoBDAV()		
for(i=0;i<N1;i++){		
padding_word	16	bslbf
}		
TableOfPlayLists()		
for(i=0;i<N2;i++){		
padding_word	16	bslbf
}		
MakersPrivateData()		
for(i=0;i<N3;i++){		
padding_word	16	bslbf
}		
}		

Фиг. 5

СИНТАКСИС	Длина данных (бит)	Мнемоническая схема
UIAppInfoBDAV(){		
length	32	unimsbf
reserved_for_future_use	16	bslbf
BDAV_character_set	8	bslbf
reserved_for_word_align	6	bslbf
BDAV_protect_flag	1	bslbf
resume_valid_flag	1	bslbf
PIN	8*4	bslbf
resume_PlayList_file_name	8*10	bslbf
ref_to_menu_thumbnail_index	16	unimsbf
BDAV_name_length	8	unimsbf
BDAV_name	8*255	bslbf
}		

Фиг. 6

СИНТАКСИС	Длина данных (бит)	Мнемоническая схема
TableOfPlayLists(){		
length	32	unimsbf
number_of_PlayLists	16	unimsbf
for(i=0;i<number_of_PlayLists,i++){		
PlayList_file_name	8*10	bslbf
}		
}		

ФИГ. 7

СИНТАКСИС	Длина данных (бит)	Мнемоническая схема
xxxxx.rpls/yyyyy.vpls{		
type_indicator	8*4	bslbf
version_number	8*4	bslbf
PlayList_start_address	32	unimsbf
PlayListMark_start_address	32	unimsbf
MakersPrivateData_start_address	32	unimsbf
reserved_for_future_use	160	bslbf
UIAppInfoPlayList()		
for(i=0;i<N1;i++){		
padding_word	16	bslbf
}		
PlayList()		
for(i=0;i<N2;i++){		
padding_word	16	bslbf
}		
PlayListMark()		
for(i=0;i<N3;i++){		
padding_word	16	bslbf
}		
MakersPrivateData()		
for(i=0;i<N4;i++){		
padding_word	16	bslbf
}		
}		

ФИГ. 8

СИНТАКСИС	Длина данных (бит)	Мнемоническая схема
UIAppInfoPlayList(){		
length	32	unimsbf
reserved_for_future_use	16	bslbf
PlayList_character_set	8	unimsbf
reserved_for_word_align	4	bslbf
playback_protect_flag	1	bslbf
write_protect_flag	1	bslbf
is_played_flag	1	bslbf
is_edited_flag	1	bslbf
time_zone	8	bslbf
reserved_for_word_align	8	bslbf
record_time_and_date	4*14	bslbf
PlayList_duration	4*6	bslbf
maker_ID	16	unimsbf
maker_model_code	16	unimsbf
channel_number	16	unimsbf
reserved_for_word_align	8	bslbf
channel_name_length	8	unimsbf
channel_name	8*20	bslbf
PlayList_name_length	8	unimsbf
PlayList_name	8*255	bslbf
PlayList_detail_length	16	unimsbf
PlayList_detail	8*1200	bslbf
}		

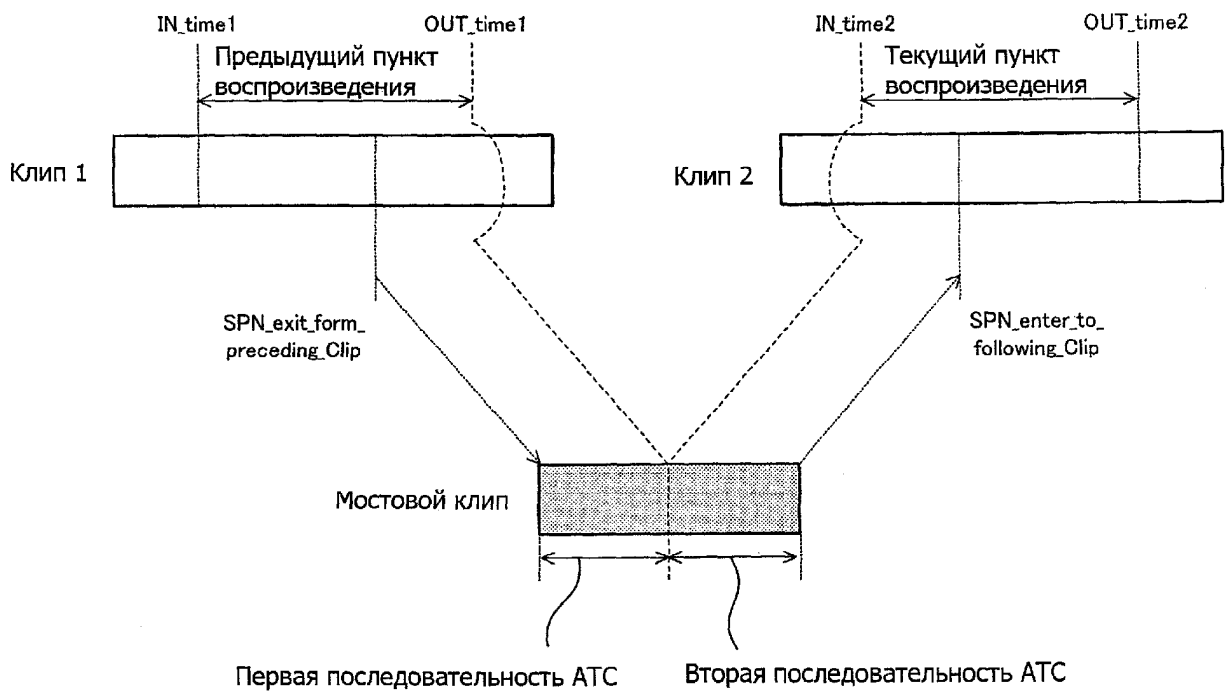
ФИГ. 9

СИНТАКСИС	Длина данных (бит)	Мнемоническая схема
PlayList(){		
length	32	unimsbf
reserved_for_word_align	12	bslbf
PL_CPI_type	4	bslbf
number_of_PlayItems	16	unimsbf
if(<<Virtual-PlayList>&&PL_CPI_type==1){		
number_of_SubPlayItems	16	unimsbf
}else{		
reserved_for_word_align	16	bslbf
}		
for(Playitem_id=0;Playitem_id<number_of_PlayItems;Playitem_id++){		
PlayItem()		
}		
if(<<Virtual-PlayList>&&CPI_type==1){		
for(i=0;i<number_of_SubPlayItems;i++){		
SubPlayItem()		
}		
}		
}		

ФИГ. 10

Синтаксис	Длина данных (бит)	Мнемоническая схема
PlayItem(){		
length	16	unimsbf
Clip_Infomation_file_name	8*5	bslbf
Clip_codec_Identifier	8*4	bslbf
reserved_for_future_use	6	bslbf
connection_condition	2	bslbf
if(CPI_type==1){		
ref_to_STC_id	8	unimsbf
}else{		
reserved_for_word_align	8	bslbf
}		
IN_time	32	unimsbf
OUT_time	32	unimsbf
if(<Virtual-PlayList>&&connection_condition==3){		
BridgeSequenceInfo()		
}		
}		

ФИГ. 11



ФИГ. 12

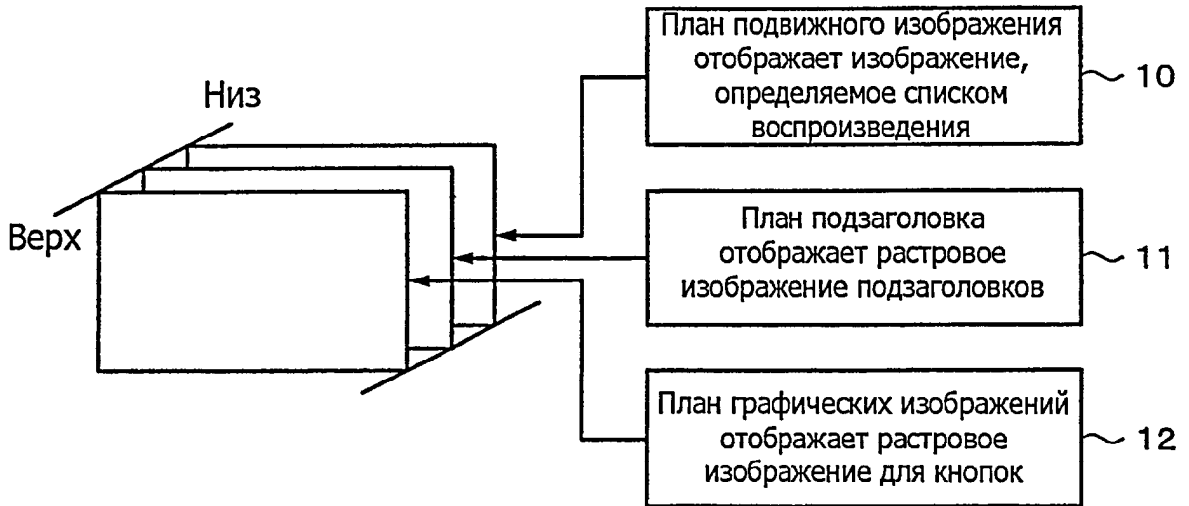


СИНТАКСИС	Длина данных (бит)	Мнемоническая схема
PlayListMark(){		
length	32	unimsbf
number_of_PlayList_marks	16	unimsbf
for(i=0;i<number_of_PlayList_marks;i++){		
mark_invalid_flag	1	unimsbf
mark_type	7	unimsbf
mark_name_length	8	unimsbf
maker_ID	16	unimsbf
ref_to_PlayItem_id	16	unimsbf
mark_time_stamp	32	unimsbf
entry_ES_PID	16	unimsbf
if(mark_type==0x01  mark_type==0x02){		
ref_to_menu_thumbnail_index	16	unimsbf
}else{		
ref_to_menu_thumbnail_index	16	unimsbf
}		
duration	32	unimsbf
makers_infomation	32	bslbf
mark_name	8*24	bslbf
}		
}		

Фиг. 13

СИНТАКСИС	Длина данных (бит)	Мнемоническая схема
zzzz.cpi{		
type_indicator	8*4	bslbf
version_number	8*4	bslbf
SequenceInfo_start_address	32	unimsbf
ProgramInfo_start_address	32	unimsbf
CPI_start_address	32	unimsbf
ClipMark_start_address	32	unimsbf
MakersPrivateData_start_address	32	unimsbf
reserved_for_future_use	96	bslbf
ClipInfo()		
for(i=0,<N1;i++){		
padding_word	16	bslbf
}		
SequenceInfo()		
for(i=0,<N2;i++){		
padding_word	16	bslbf
}		
ProgramInfo()		
for(i=0,<N3;i++){		
padding_word	16	bslbf
}		
CPI()		
for(i=0,<N4;i++){		
padding_word	16	bslbf
}		
ClipMark()		
for(i=0,<N5;i++){		
padding_word	16	bslbf
}		
MakersPrivateData()		
for(i=0,<N6;i++){		
padding_word	16	bslbf
}		
}		

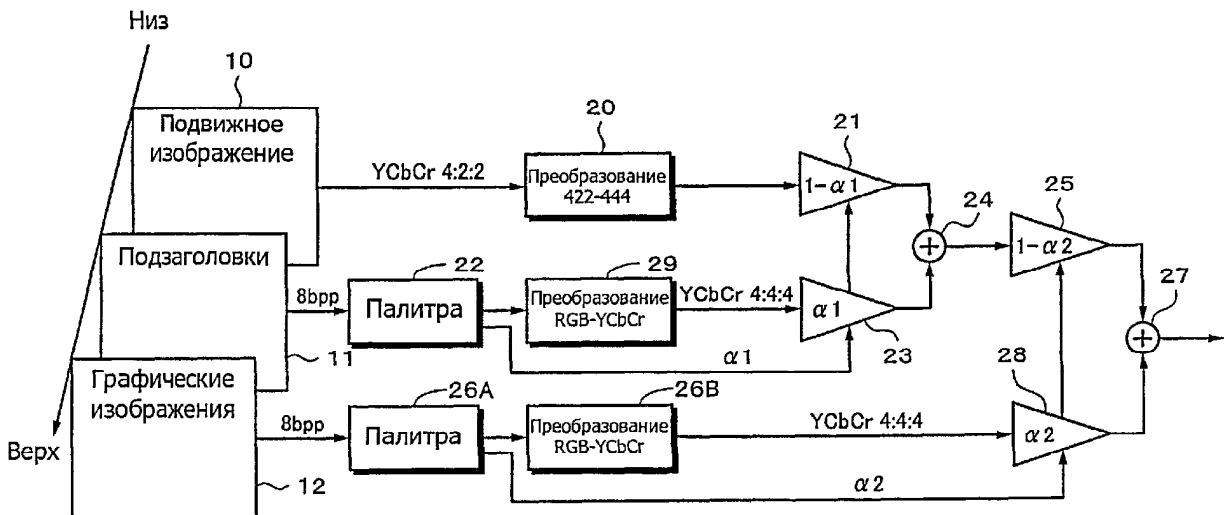
Фиг. 14



ФИГ. 15

Пункт	Описание
План подвижного изображения	1920 x 1080 x 16 битов, YCbCr (4 : 2 : 2), 8 битов каждый
План подзаголовка	1920 x 1080 x 8 битов, адреса 8-битовой карты цвета (палитра) + альфа-смещение на 256 уровнях
План графического изображения	1920 x 1080 x 8 битов, адреса 8-битовой карты цвета (палитра) + альфа-смещение на 256 уровнях

ФИГ. 16



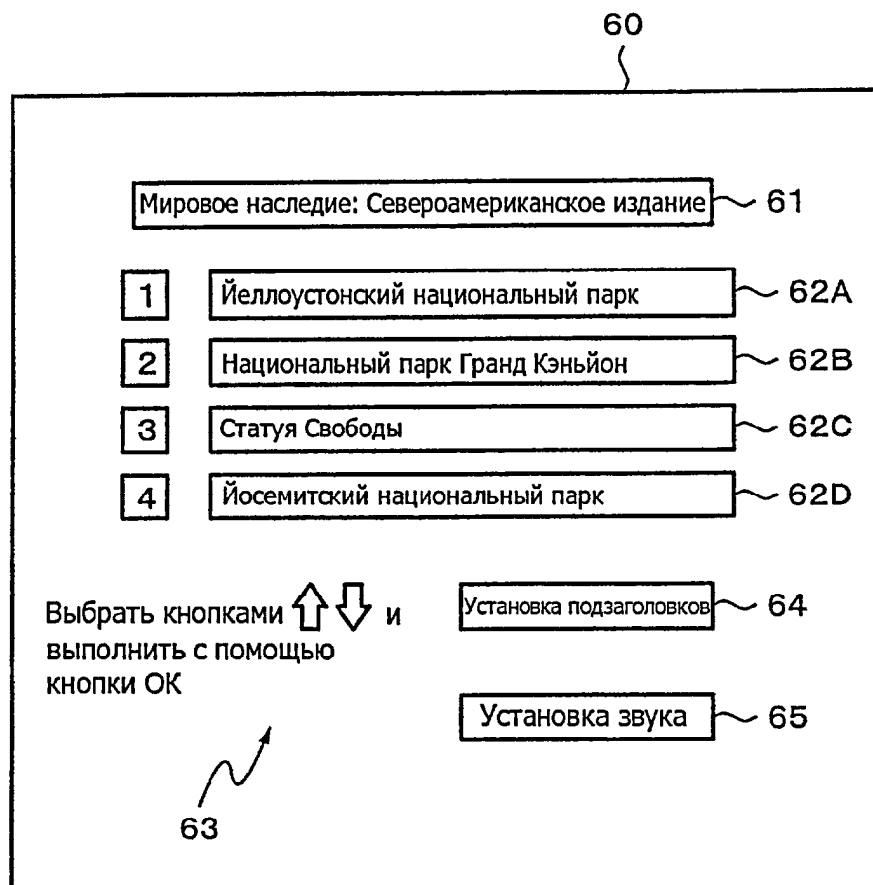
ФИГ. 17

Вход	Входной адрес, 8 бит
Выход	Выходные данные, 8 бит x 4, (R, G, B, $\alpha$ )

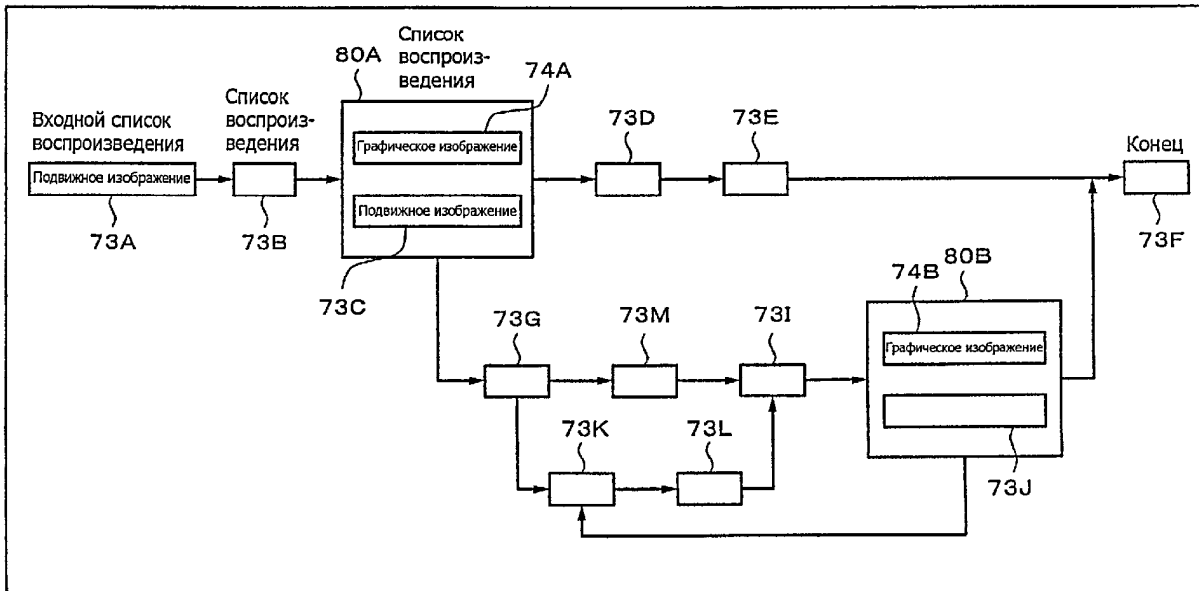
ФИГ. 18

Величина индекса цвета	Величины трех основных цветов			Степень прозрачности
	R	G	B	
0x00	0	0	0	0
0x01	10	100	30	0.5
⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮
0xFF	200	255	100	0.8

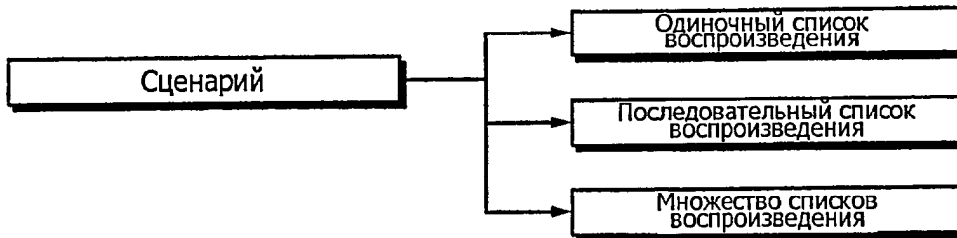
Фиг. 19



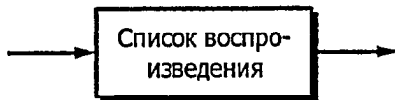
Фиг. 20



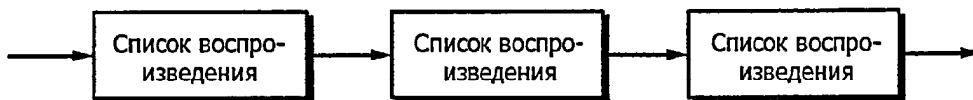
ФИГ. 21



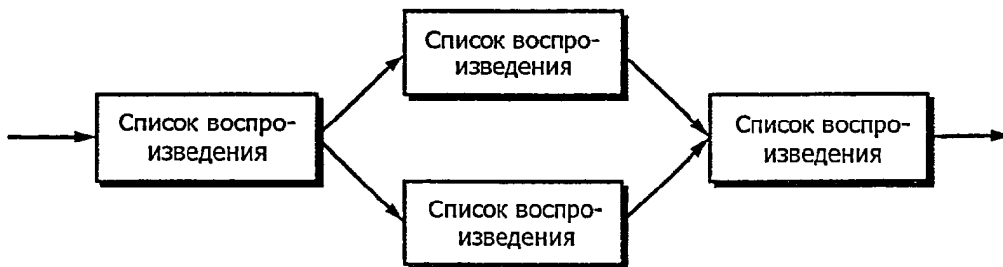
ФИГ. 22



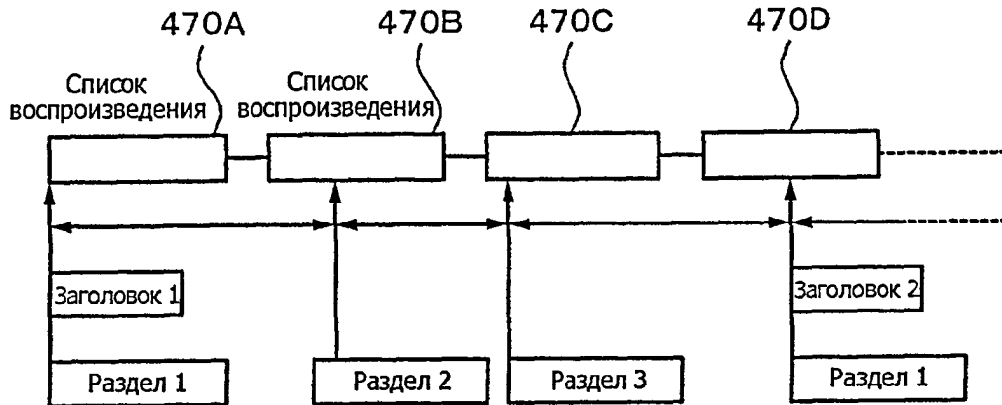
ФИГ. 23.A



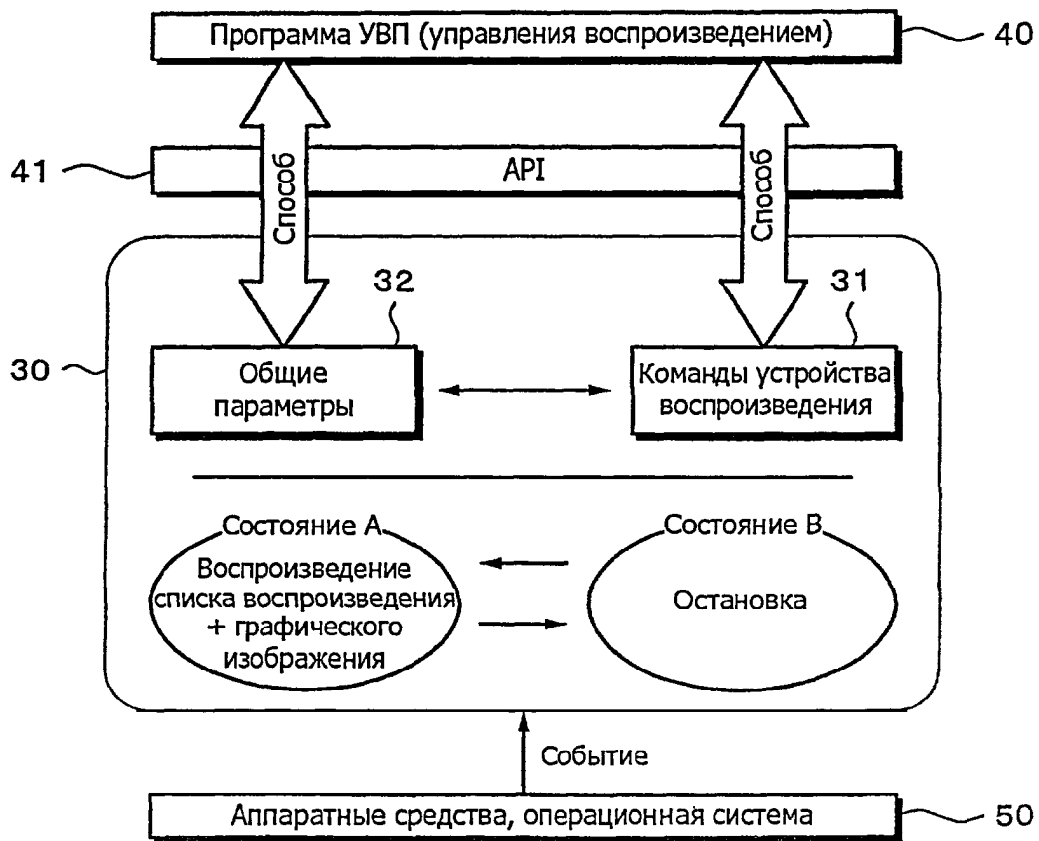
ФИГ. 23B



ФИГ. 23C



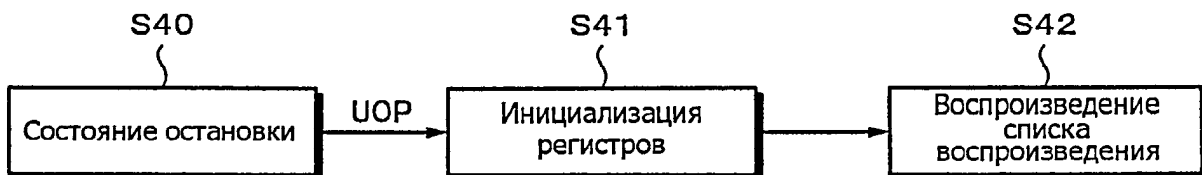
Фиг. 24



Фиг. 25

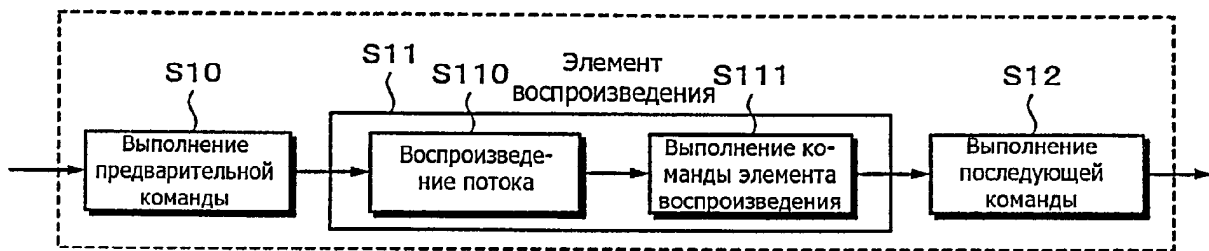


Фиг. 26А



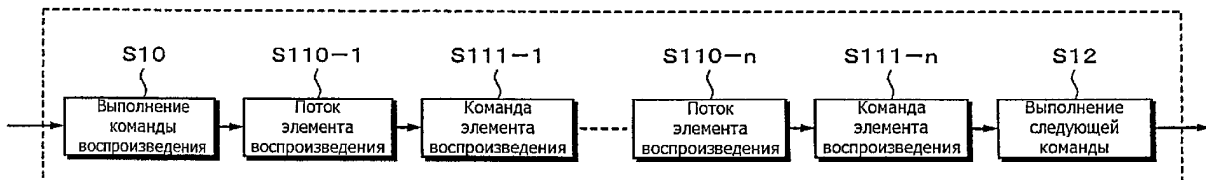
Фиг. 26В

Список воспроизведения (один элемент воспроизведения)



Фиг. 27А

Список воспроизведения (последовательные элементы воспроизведения)



Фиг. 27В

Фиг. 28А
Фиг. 28В
Фиг. 28С
Фиг. 28D
Фиг. 28Е
Фиг. 28F
Фиг. 28G
Фиг. 28H

Фиг. 28

Способ	Замечания
Команды, предназначенные для определения положения начала воспроизведения	
LinkPlayList(playListNumber)	Обеспечивает начало воспроизведения списка воспроизведения, определяемого PlayListNumber
LinkPlayItem(playListNumber,playItemNumber)	Обеспечивает начало воспроизведения обозначенного элемента воспроизведения, обозначенного списка воспроизведения. playItemNumber ПРЕДСТАВЛЯЕТ СОБОЙ playItem_id, НАЧИНАЮЩИЙСЯ С 0 КОГДА PlayList ВОСПРОИЗВОДИТСЯ С НАЧАЛА, playItemNumber РАВЕН 0

Фиг. 28А

Способ	Замечания
Команды, предназначенные для определения положения начала воспроизведения	
Link(position)(object) position=(“prev” “next” “top” “Parent” “tail”) object=(PlayList   PlayItem   Chapter)	Обеспечивает перемещение текущего положения в сценарии. Обеспечивает перемещение текущего положения воспроизведения в соседний список воспроизведения, элемент воспроизведения или раздел.
Exit	Обеспечивает остановку воспроизведения сценария. Значение стандартного регистра не сохраняется.
RSM	Обеспечивает возобновление воспроизведения от последнего положения остановки воспроизведения. Обеспечивает вызов записанной информации продолжения для установки в регистр и начала воспроизведения сценария.

Фиг. 28B

Команды для получения состояния устройства воспроизведения	
getMenuDescriptionLanguage()	Обеспечивает получение языка отображаемого меню
getScenarioNumber()	Обеспечивает получение номера воспроизводимого сценария
getPlayListNumber()	Обеспечивает получение воспроизводимого номера списка воспроизведения
getChapterNumber()	Обеспечивает получение воспроизводимого номера раздела
getPlayerSupport()	Обеспечивает получение версии и функции устройства воспроизведения

Фиг. 28C

КОМАНДЫ ДЛЯ ВИДЕОПОТОКОВ	
getVideoStreamAvailability()	ОБЕСПЕЧИВАЕТ ПОЛУЧЕНИЕ ИНФОРМАЦИИ, КОТОРАЯ ОПИСЫВАЕТ, ДЕЙСТВИТЕЛЬНО ЛИ НАЗНАЧЕННЫЙ ВИДЕОПОТОК СОДЕРЖИТСЯ
setVideoStreamNumber()	ОПИСЫВАЕТ ВИДЕОПОТОК, ПРЕДНАЗНАЧЕННЫЙ ДЛЯ ДЕКОДИРОВАНИЯ
getVideoStreamNumber()	ОБЕСПЕЧИВАЕТ ПОЛУЧЕНИЕ НОМЕРА ВЫБИРАЕМОГО ВИДЕОПОТОКА
getVideoStreamAttribute()	ОБЕСПЕЧИВАЕТ ПОЛУЧЕНИЕ АТТРИБУТА ВИДЕОПОТОКА (СИСТЕМЫ КОДИРОВАНИЯ), РАЗРЕШАЮЩЕЙ СПОСОБНОСТИ, СООТНОШЕНИЯ РАЗМЕРОВ ИЗОБРАЖЕНИЯ, РЕЖИМА ОТОБРАЖЕНИЯ В СЛУЧАЕ СООТНОШЕНИЯ РАЗМЕРОВ ИЗОБРАЖЕНИЯ 4:3, ЗАКРЫТАЯ НАДПИСЬ)
setAngleNumber()	ОПИСЫВАЕТ НОМЕР УГЛА
getAngleNumber()	ОБЕСПЕЧИВАЕТ ПОЛУЧЕНИЕ ВЫБИРАЕМОГО НОМЕРА УГЛА
getMaxVideoStreams()	ОБЕСПЕЧИВАЕТ ПОЛУЧЕНИЕ ВИДЕОПОТОКОВ, КОТОРЫЕ МОГУТ БЫТЬ ВЫБРАНЫ ОПИСЫВАЕТ, ДЕЙСТВИТЕЛЬНО ЛИ getVideoStreamAvailability () ЯВЛЯЕТСЯ ДОСТАТОЧНЫМ, ИЛИ НЕТ

Фиг. 28D



Команды для звуковых потоков	
getAudioStreamAvailability()	Обеспечивает получение информации, которая описывает, действительно ли назначенный видеопоток содержит
getAudioStreamLanguage()	Обеспечивает получение информации о языке обозначенного звукового потока
getAudioStreamStatus()	Описывает воспроизводимый звуковой поток
setAudioStreamStatus()	Обеспечивает получение номера воспроизводимого звукового потока
getAudioStreamAttribute()	Обеспечивает получение атрибута звукового потока (система кодирования количества каналов Q, FS)

Фиг. 28E

Команды для потоков подизображения	
getSPStreamAvailability()	Обеспечивает получение информации, которая описывает, действительно ли содержится обозначенный поток SP
getSPStreamLanguage()	Обеспечивает получение языка обозначенного SP потока
getSPDisplayStatus()	Обеспечивает получение состояния отображения SP (отображается SP или нет)
setSPDisplayStatus()	Описывает состояние отображения SP (действительно ли отображается SP)
getSpStreamAttribute()	Обеспечивает получение атрибута SP (разрешение 4:3 или широкий экран)

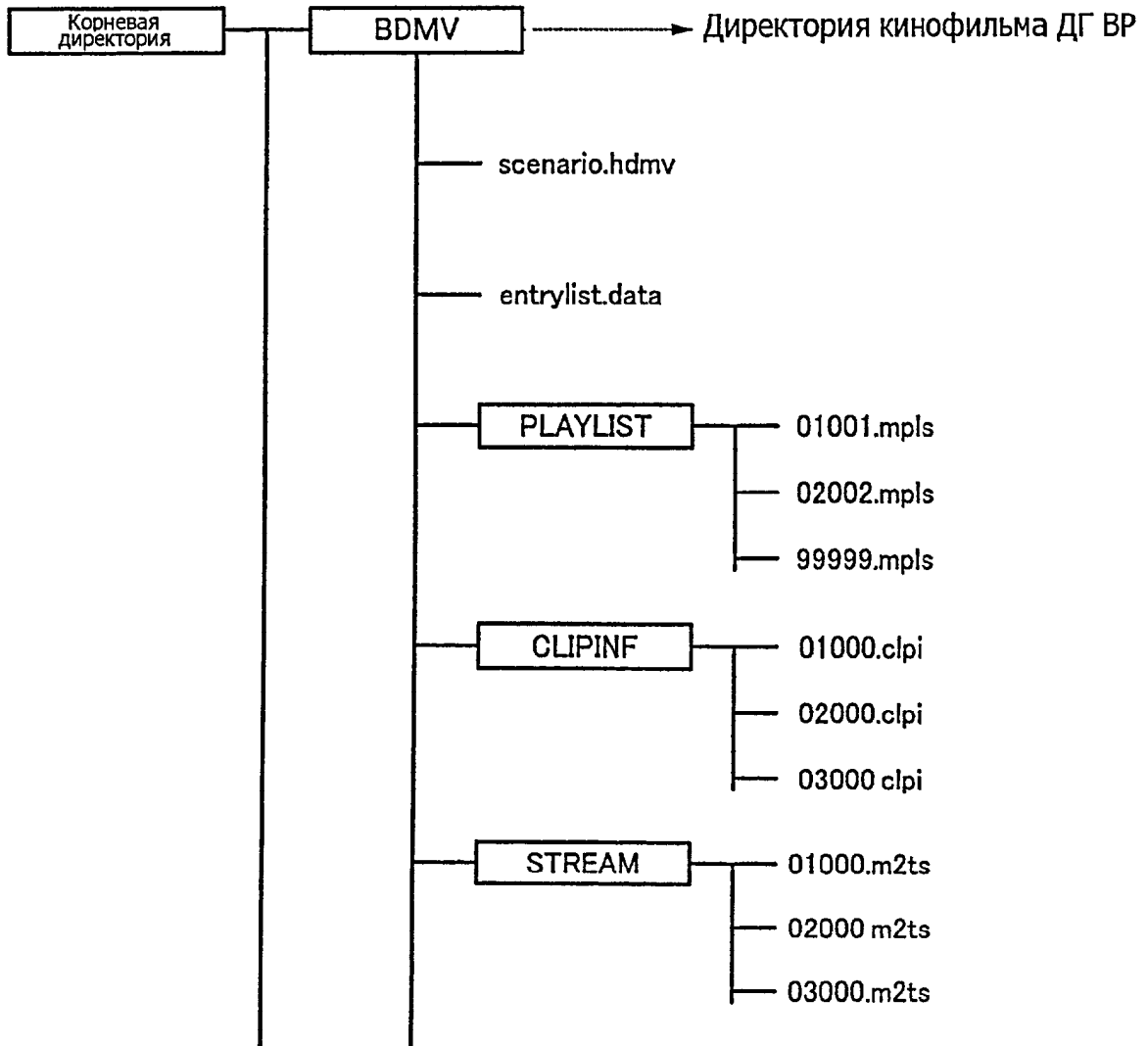
Фиг. 28F

Команды для регистра считывания/записи	
clearReg()	Обеспечивает инициализацию всех регистров
setReg()	Обеспечивает установку значения в регистр
getReg()	Обеспечивает считывание значения из регистра

Фиг. 28G

Команды для таймеров	
sleep()	Обеспечивает остановку процесса для обозначения миллисекунд
setTimeout()	Обеспечивает выполнение функции и процесса после того, как пройдут определенные миллисекунды
setInterval()	Обеспечивает выполнение процесса на обозначенных миллисекундных интервалах
clearTimer()	Обеспечивает остановку процесса таймера, который имеет определенный идентификатор регистрации таймера
pauseTimer()	Обеспечивает временную остановку таймера, который имеет определенный идентификатор регистрации таймера
resumeTimer()	Обеспечивает продолжение работы таймера, который имеет определенный идентификатор таймера регистрации, после состояния паузы
Другие команды	
playSoundEffect(sound_id)	Обеспечивает воспроизведение выбранного звукового эффекта Используется с командной кнопки

Фиг. 28H



Фиг. 29

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
scenario.hdmv{		
type_indicator	8*4	bslbf
version_number	8*4	bslbf
scenario_start_address	32	
reserved_for_future_use	224	bslbf
Autoplay()		
for(i=0,i<N1;i++){		
padding_word	16	bslbf
}		
Scenario()		
for(i=0,i<N2;i++){		
padding_word	16	bslbf
}		
}		

Фиг. 30

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
Autoplay(){		
length	32	uimsbf
reserved	16	
number_of_commands	16	
for(i=0,i<number_of_commands,i++){		
command(i)	32	uimsbf
}		
}		

Фиг. 31

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
Scenario(){		
length	32	
flags	32	
number_of_PlayLists	16	
for(i=0;i<Number_of_PlayLists;i++){		
Pre_Command_start_id	32	
Post_Command_start_id	32	
number_of_Pre_Commands	32	
number_of_Post_Commands	32	
reserved	32	
number_of_PlayItems	32	
for(PlayItem_id=0,PlayItem_id<number_of_PlayItems;PlayItem_id++){		
PI_Command_start_id	32	
number_of_PI_Commands	32	
}		
reserved		
// Command table for each Playlist		
number_of_PL_Commands	16	
for(j=0;j<number_of_PL_Commands;j++){		
PL_Command(j)	32	
}		
}		

Фиг. 32

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
entrylist.data{		
type_indicator	8*4	bslbf
version_number	8*4	bslbf
ScenarioEntry_start_address	32	uimsbf
reserved_for_future_use	224	bslbf
AppInfo()		
for(i=0;i<N1;i++){		
padding_word	16	bslbf
}		
ScenarioEntry()		
for(i=0;i<N2;i++){		
padding_word	16	bslbf
}		
}		

Фиг. 33

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
AppInfo(){		
length	32	uimsbf
reserved_for_future_use	16	bslbf
HDMV_name_character_set	8	bslbf
reserved_for_word_align	7	bslbf
PIN_valid_flag	1	bslbf
PIN	8*4	bslbf
//    UOP_mask_table() // For directory	64	
HDMV_name_length	8	uimsbf
HDMV_name	8*255	bslbf
}		

Фиг. 34

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
ScenarioEntry(){		
length	32	uimsbf
name_character_set	8	bslbf
//    Entry PL for the Top Menu		
Top Menu PL(){		
flags	32	bslbf
TopMenu_ref_to_PlayList_file_name	8*10	bslbf
TopMenu_ref_to_PlayItem_id	16	uimsbf
TopMenu_name_length	8	uimsbf
TopMenu_name	8*255	bslbf
}		
//    Title Entries		
number_of_Titles	16	uimsbf
for(unit title_number=0;title_number<Number_of_Titles;title_number++){		
flags	32	bslbf
Title_ref_to_PlayList_file_name	8*10	bslbf
Title_ref_to_PlayItem_id	16	uimsbf
Title_name_length	8	uimsbf
Title_name	8*255	bslbf
}		
//    Stream Setup Menu for each PL		
number_of_PlayLists	16	uimsbf
for(i=0;i<Number_of_PlayLists;i++){		
SSMenu_flags	32	bslbf
SSMenu_ref_to_PlayList_file_name	8*10	bslbf
SSMenu_ref_to_PlayItem_id	16	uimsbf
}		
}		

Фиг. 35

Синтаксис	Длина данных (битов)	Мнемоническое обозначение
xxxxx.mpls{		
type_indicator	8*4	bslbf
version_number	8*4	bslbf
PlayList_start_address	32	unimsbf
PlayListMark_start_address	32	unimsbf
reserved_for_future_use	192	bslbf
PLControlInfo()		
for(i=0;i<N1;i++){		
padding_word	16	bslbf
}		
PlayList()		
for(i=0;i<N2;i++){		
padding_word	16	bslbf
}		
PlayListMark()		
for(i=0;i<N3;i++){		
padding_word	16	bslbf
}		
}		

ФИГ. 36

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
PLControlInfo{		
length	32	unimsbf
reserved_for_future_use	8	bslbf
PlayList_character_set	8	unimsbf
reserved_for_future_use	8	
PL_playback_type	8	
if(PL_playback_type==0x2		
PL_playback_type==0x3) {		
playback_count	16	
} else {		
reserved_for_word_align	16	
}		
PL_UOP_mask_table() // For PlayList	64	
reserved_for_word_align	8	
PL_random_access_mode	8	
reserved_for_word_align	8	bslbf
PlayList_duration	4*6	bslbf
PlayList_name_length	8	unimsbf
PlayList_name	8*255	bslbf
PlayList_detail_length	16	unimsbf
PlayList_detail	8*1200	bslbf
}		

Фиг. 37

PL_playback_type	Описание
0x0	Зарезервированная область
0x1	Последовательно воспроизводит элементы воспроизведения (нормальное воспроизведение)
0x2	Случайно воспроизводит элементы воспроизведения
0x3	Воспроизводит элементы воспроизведения со случайной перестановкой

Фиг. 38

PL_random_access_mode	Описание
0x0	Обеспечивает воспроизведение с переходом и воспроизведение с переменной скоростью
0x1	Запрещает воспроизведение с переходом и воспроизведение с переменной скоростью

Фиг. 39

Синтаксис	Длина данных (битов)	Мнемоническое обозначение
PlayList(){		
length	32	unimsbf
number_of_PlayItems	16	unimsbf
number_of_SubPlayItems	16	unimsbf
for(PlayItem_id=0;PlayItem_id<number_of_PlayItems;PlayItem_id++){		
PlayItem()		
}		
for(SubPlayItem_id=0;SubPlayItem_id<number_of_SubPlayItems;SubPlayItem_id++){		
SubPlayItem()		
}		
}		

ФИГ. 40



СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
PlayItem(){		
length	16	uimsbf
reserved_for_word_align	8	bslbf
Clip_Information_file_name	8*5	bslbf
Clip_codec_identifier	8*4	bslbf
reserved_for_future_use	7	bslbf
is_multi_angle	1	bslbf
reserved_for_future_use	4	bslbf
connection_condition	4	uimsbf
ref_to_STC_id	8	uimsbf
IN_time	32	uimsbf
OUT_time	32	uimsbf
PI_UOP_mask_table()	64	bslbf
PID_filter()		
reserved_for_word_align	8	bslbf
PI_random_access_mode	8	uimsbf
reserved_for_word_align	8	bslbf
still_mode	8	uimsbf
if(still_mode==0x1){		
still_time	16	uimsbf
} else {		
reserved_for_word_align	16	bslbf
}		
// Angle		
if(is_multi_angle){		
number_of_angles	8	uimsbf
is_seamless_angle_change	8	uimsbf
for(angle_id=1,angle_id<number_of_angles,angle_id++){		
Clip_Information_file_name	8*5	bslbf
ref_to_STC_id	8	uimsbf
IN_time	32	uimsbf
OUT_time	32	uimsbf
}		
}		
}		

ФИГ. 41

PI_random_access_mode	Описание
0x0	Обеспечивает воспроизведение с переходом и воспроизведение с переменной скоростью
0x1	Запрещает воспроизведение с переходом и воспроизведение с переменной скоростью

Фиг. 42

still_mode	Описание
0x0	Отсутствует неподвижное изображение
0x1	Неподвижное изображение в течение ограниченного периода времени. Период времени описан в still_time.
0x2	Неподвижное изображение в течение ограниченного периода времени. Неподвижное изображение продолжается до тех пор, пока пользователь его не отменит.
0x3-0xf	Зарезервировано

Фиг. 43

is_seamless_angle_change	Описание
0x0	Возможность небезстыкового изменения углов
0x1	Возможность безстыкового изменения углов

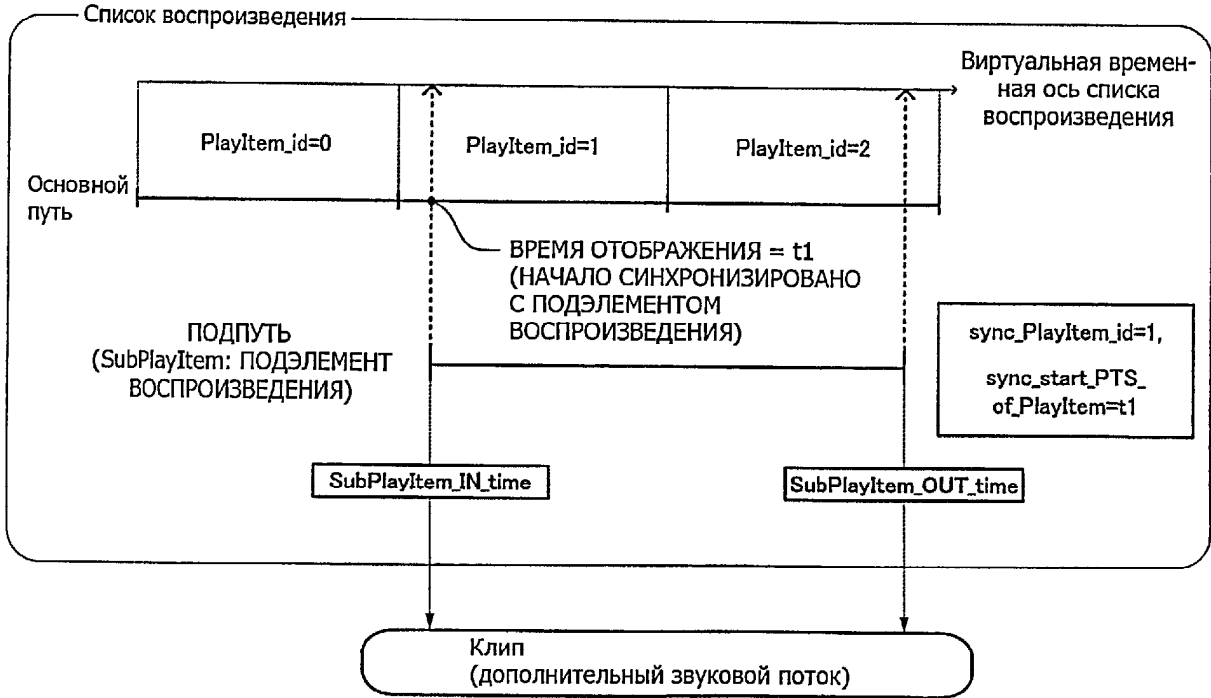
Фиг. 44

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
SubPlayItem() {		
length	16	unimsbf
Clip_Information_file_name	8*5	bslbf
Clip_codec_identifier	8*4	bslbf
reserved_for_future_use	7	bslbf
is_repeat_flag	1	bslbf
SubPlayItem_type	8	bslbf
ref_to_STC_id	8	unimsbf
SubPlayItem_IN_time	32	unimsbf
SubPlayItem_OUT_time	32	unimsbf
if(is_repeat_flag==0) {		
sync_PlayItem_id	16	unimsbf
sync_start_PTS_of_PlayItem	32	unimsbf
} else {		
reserved_for_word_align	16	
reserved_for_word_align	32	
}		
}		

Фиг. 45

is_repeat_flag	Описание
0	Выполняет воспроизведение синхронно с основным путем
1	Не выполняет воспроизведение синхронно с основным путем. Повторяет воспроизведение.

ФИГ. 46



ФИГ. 47

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
zzzzz clipi {		
type_indicator	8*4	bslbf
version_number	8*4	bslbf
SequenceInfo_start_address	32	uimsbf
ProgramInfo_start_address	32	uimsbf
CPI_start_address	32	uimsbf
ClipMark_start_address	32	uimsbf
reserved_for_future_use	128	bslbf
ClipInfo()		
for(i=0; i<N1; i++){		
padding_word	16	bslbf
}		
SequenceInfo()		
for(i=0; i<N2; i++){		
padding_word	16	bslbf
}		
ProgramInfo()		
for(i=0; i<N3; i++){		
padding_word	16	bslbf
}		
CPI()		
for(i=0; i<N4; i++){		
padding_word	16	bslbf
}		
ClipMark()		
for(i=0; i<N5; i++){		
padding_word	16	bslbf
}		
}		

Фиг. 48

Синтаксис	Длина данных (битов)	Мнемоническое обозначение
ClipInfo(){		
length	32	unimsbf
reserved	8	bslbf
application_type	8	unimsbf
Clip_stream_type	8	unimsbf
reserved	40	unimsbf
TS_recording_rate	32	unimsbf
num_of_source_packets	32	unimsbf
BD_system_use	1024	bslbf
TS_type_info_block()		
}		

Фиг. 49

application_type	Описание
0	СООТВЕТСТВУЮЩИЙ ФАЙЛ m2ts НЕ СООТВЕТСТВУЕТ ПРАВИЛУ ТРАНСПОРТНОГО ПОТОКА HDMV.
1	СООТВЕТСТВУЮЩИЙ ФАЙЛ m2ts СООТВЕТСТВУЕТ ПРАВИЛУ ТРАНСПОРТНОГО ПОТОКА HDMV. (НОРМАЛЬНЫЙ ПОТОК HDMV)
2	СООТВЕТСТВУЮЩИЙ ФАЙЛ m2ts СООТВЕТСТВУЕТ ПРАВИЛУ ТРАНСПОРТНОГО ПОТОКА HDMV ДЛЯ НЕПОДВИЖНОГО ИЗОБРАЖЕНИЯ, КОТОРОЕ СИНХРОНИЗИРОВАНО С ВОСПРОИЗВЕДЕНИЕМ ЗВУКА. (ДЕМОНСТРАЦИЯ СЛАЙДОВ НА ВРЕМЕННОЙ ОСИ)
3	СООТВЕТСТВУЮЩИЙ ФАЙЛ m2ts СООТВЕТСТВУЕТ ПРАВИЛУ ТРАНСПОРТНОГО ПОТОКА HDMV ДЛЯ НЕПОДВИЖНОГО ИЗОБРАЖЕНИЯ, КОТОРОЕ ВОСПРОИЗВОДЯТ НЕСИНХРОННО СО ЗВУКОМ. (ДЕМОНСТРАЦИЯ СЛАЙДОВ С ВОЗМОЖНОСТЬЮ ПРОИЗВОЛЬНОГО ПРОСМОТРА)

Фиг. 50

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
SequenceInfo(){		
length	32	unimsbf
reserved_for_word_align	8	bslbf
num_of_ATC_sequences	8	unimsbf
for(ato_id=0;ato_id<num_of_ATC_sequences;ato_id++){		
SPN_ATO_start[ato_id]	32	unimsbf
num_of_STC_sequences[ato_id]	8	unimsbf
offset_STC_id[ato_id]	8	unimsbf
for(stc_id=offset_STC_id[ato_id],stc_id<(num_of_STC_sequences[ato_id]+offset_STC_id[ato_id]);stc_id++){		
PGR_PID[ato_id][stc_id]	16	unimsbf
SPN_STC_start[ato_id][stc_id]	32	unimsbf
presentation_start_time[ato_id][stc_id]	32	unimsbf
presentation_end_time[ato_id][stc_id]	32	unimsbf
}		
}		
}		

Фиг. 51

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
ProgramInfo(){		
length	32	unimsbf
reserved_for_word_align	8	bslbf
num_of_program_sequences	8	unimsbf
for(i=0;i<num_of_program_sequences;i++){		
SPN_program_sequence_start[i]	32	unimsbf
program_map_PID[i]	16	bslbf
num_of_streams_in_ps[i]	8	unimsbf
num_of_groups[i]	8	unimsbf
for(stream_index=0,stream_index<num_of_streams_in_ps[i];stream_index++){		
stream_PID[i][stream_index]	16	unimsbf
StreamCodingInfo(i,stream_index)		
}		
}		
}		

Фиг. 52

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
StreamCodingInfo(i,stream_index){		
length	8	bslbf
stream_coding_type	8	unimsbf
if(stream_coding_type==0x02){		
video_format	4	unimsbf
frame_rate	4	unimsbf
aspect_ratio	4	unimsbf
reserved_for_word_align	2	bslbf
cc_flag	1	unimsbf
reserved_for_word_align	1	bslbf
}else if(stream_coding_type==0x80//stream_coding_type==0x81//stream_coding_type==0x82){		
audio_presentation_type	4	unimsbf
sampling_frequency	4	unimsbf
language_code	16	bslbf
reserved_for_word_align	8	bslbf
}else if(stream_coding_type==0x90){		
language_code	16	bslbf
T.B.D		
}else if(stream_coding_type==0xA0 ){}		
language_code	16	bslbf
T.B.D		
}		
}		

Фиг. 53

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
CPI(){		
length	32	unimsbf
reserved_for_word_align	12	bslbf
CPI_type	4	unimsbf
EP_map_for_BDMV()		
}		

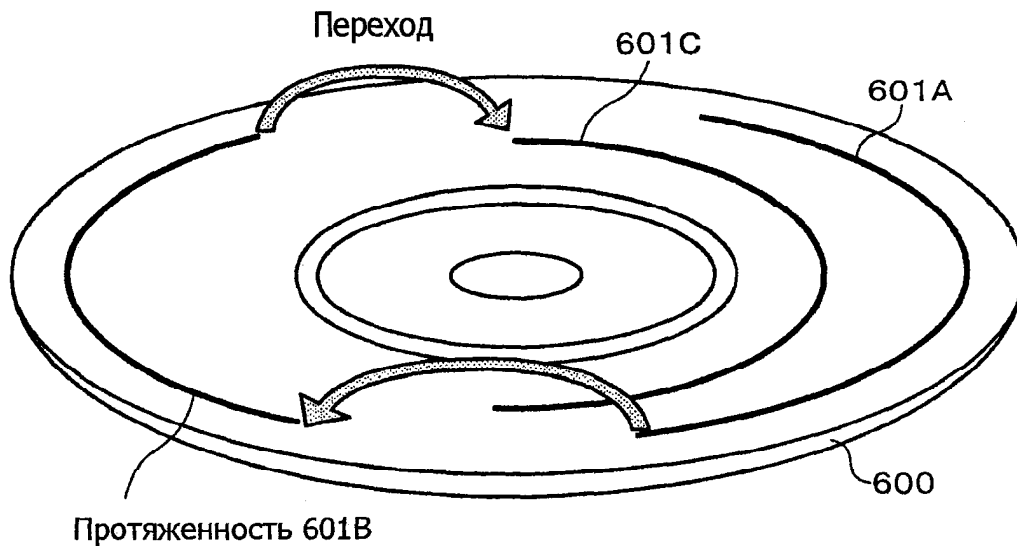
Фиг. 54

CPI_type	Описание
0	Зарезервировано для использования в будущем
1	Тип EP_map
2	Тип TU_map
3-7	Зарезервировано для использования в будущем
8	ТИП EP_map ДЛЯ BDMV
9-15	Зарезервировано для использования в будущем

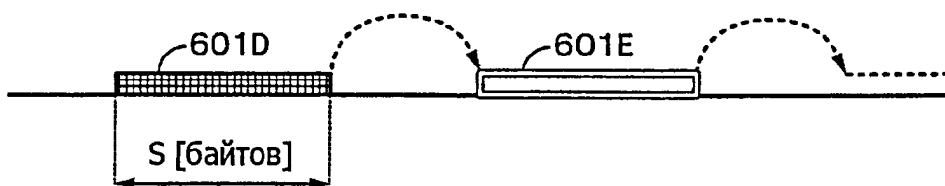
Фиг. 55

СИНТАКСИС	Длина данных (битов)	Мнемоническое обозначение
EP_map_for_one_stream_PID(EP_stream_type, Nc, Nf){		
EP_fine_table_start_address	32	unimsbf
for(i=0; i<Nc; i++){		
ref_to_EP_fine_id[i]	18	unimsbf
PTS_EP_coarse[i]	14	unimsbf
SPN_EP_coarse[i]	32	unimsbf
}		
for(i=0; i<X; i++){		
padding_word	16	bslbf
}		
for(EP_fine_id=0; EP_fine_id<Nf; EP_fine_id++){		
is_angle_change_point[EP_fine_id]	1	bslbf
Lend_position_offset[EP_fine_id]	3	bslbf
PTS_EP_fine[EP_fine_id]	11	unimsbf
SPN_EP_fine[EP_fine_id]	17	unimsbf
}		
}		

Фиг. 56

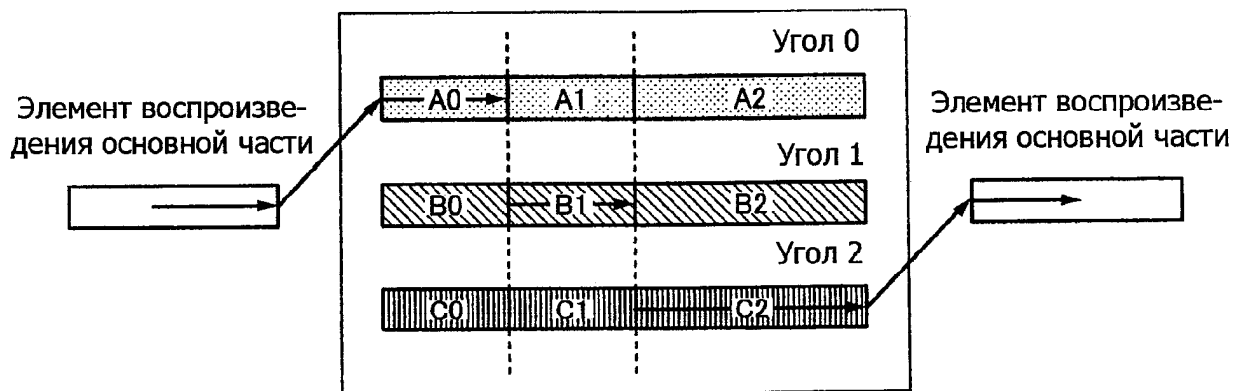


Фиг. 57

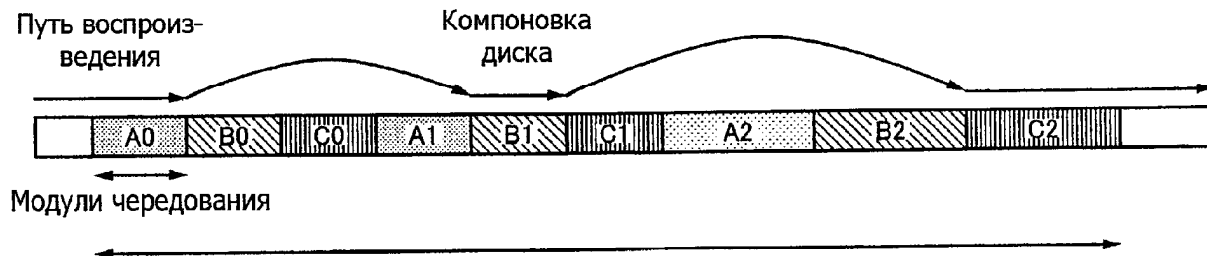


Фиг. 58

Блок множества углов

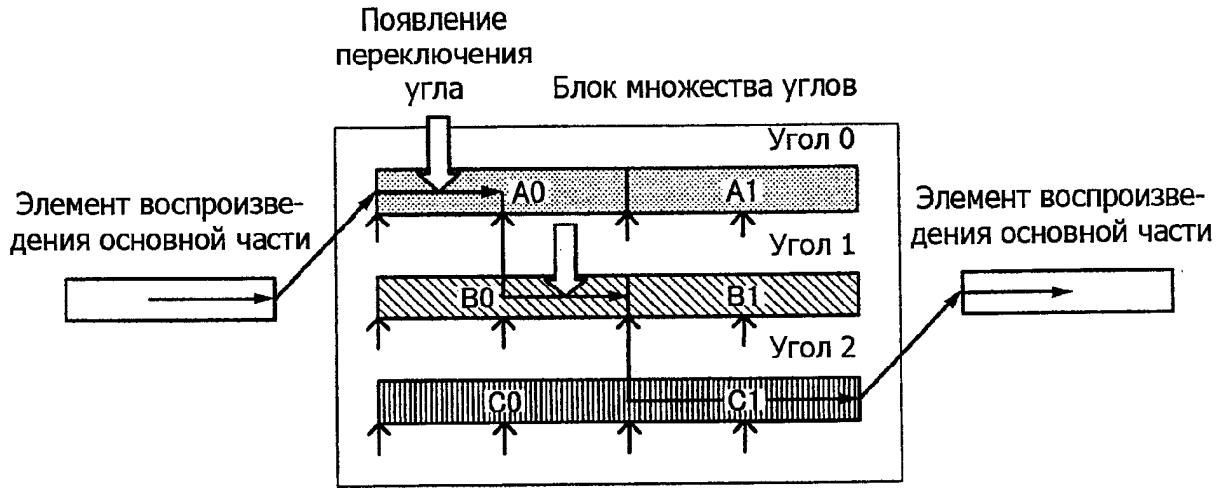


Фиг. 59А

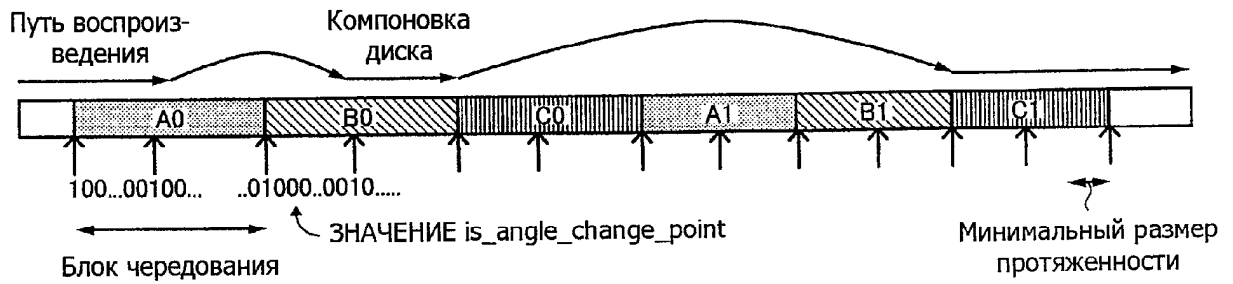


Фиг. 59В





Фиг. 60А

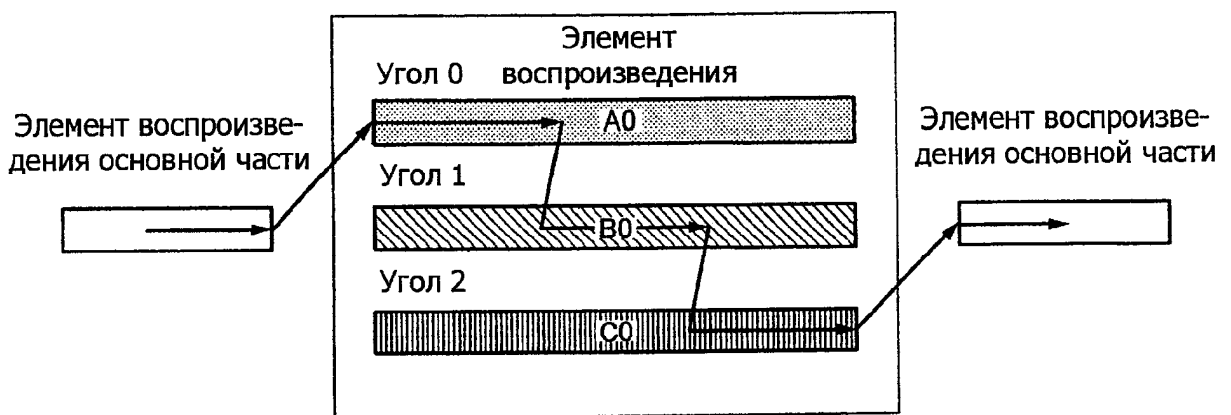


Фиг. 60В

is_angle_change_point	Значение
0	Этот вход ТВ не соответствует точке возможности переключения угла
1	Вход ТВ соответствует точке возможности переключения угла

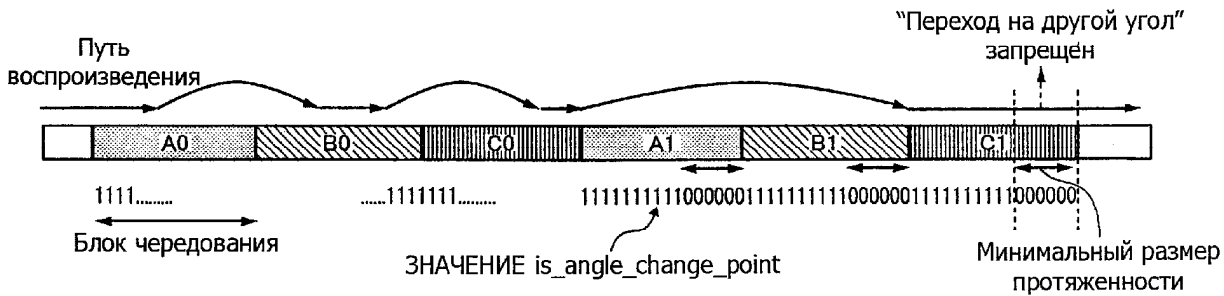
Фиг. 61

Блок множества углов

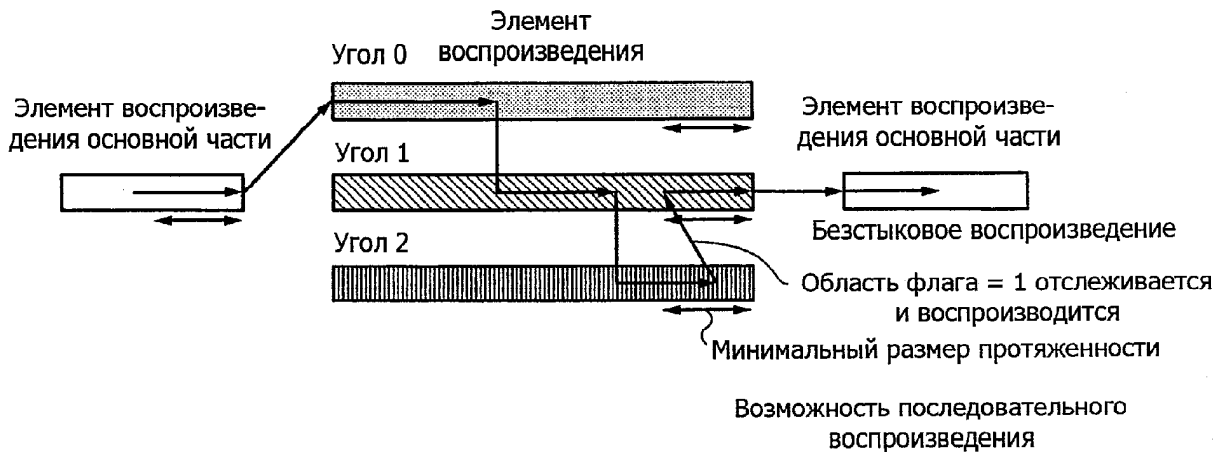


Фиг. 62А

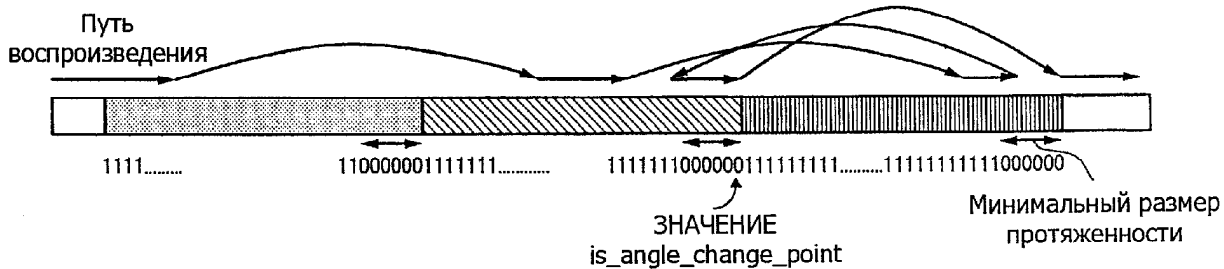




Фиг. 64В



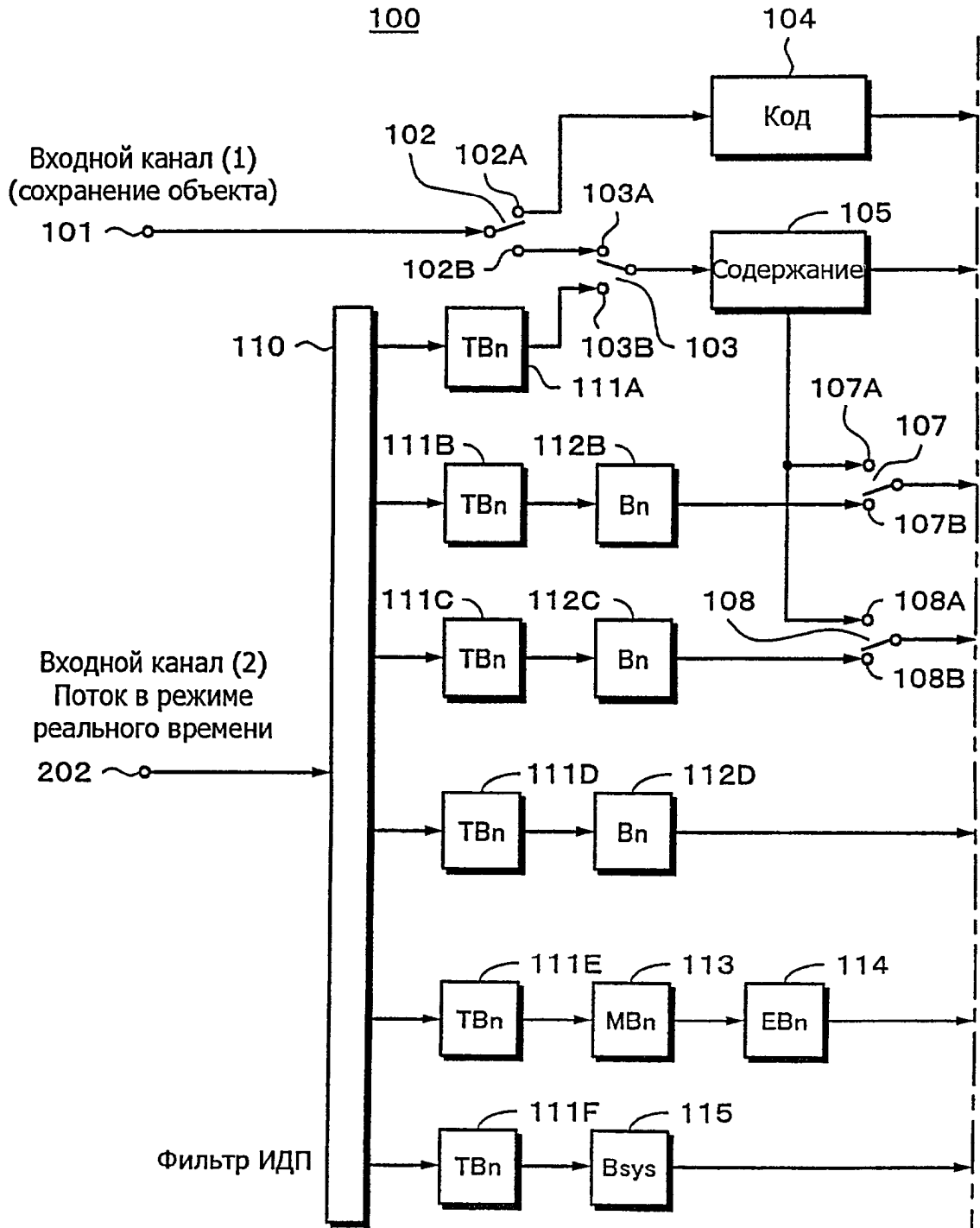
Фиг. 65А



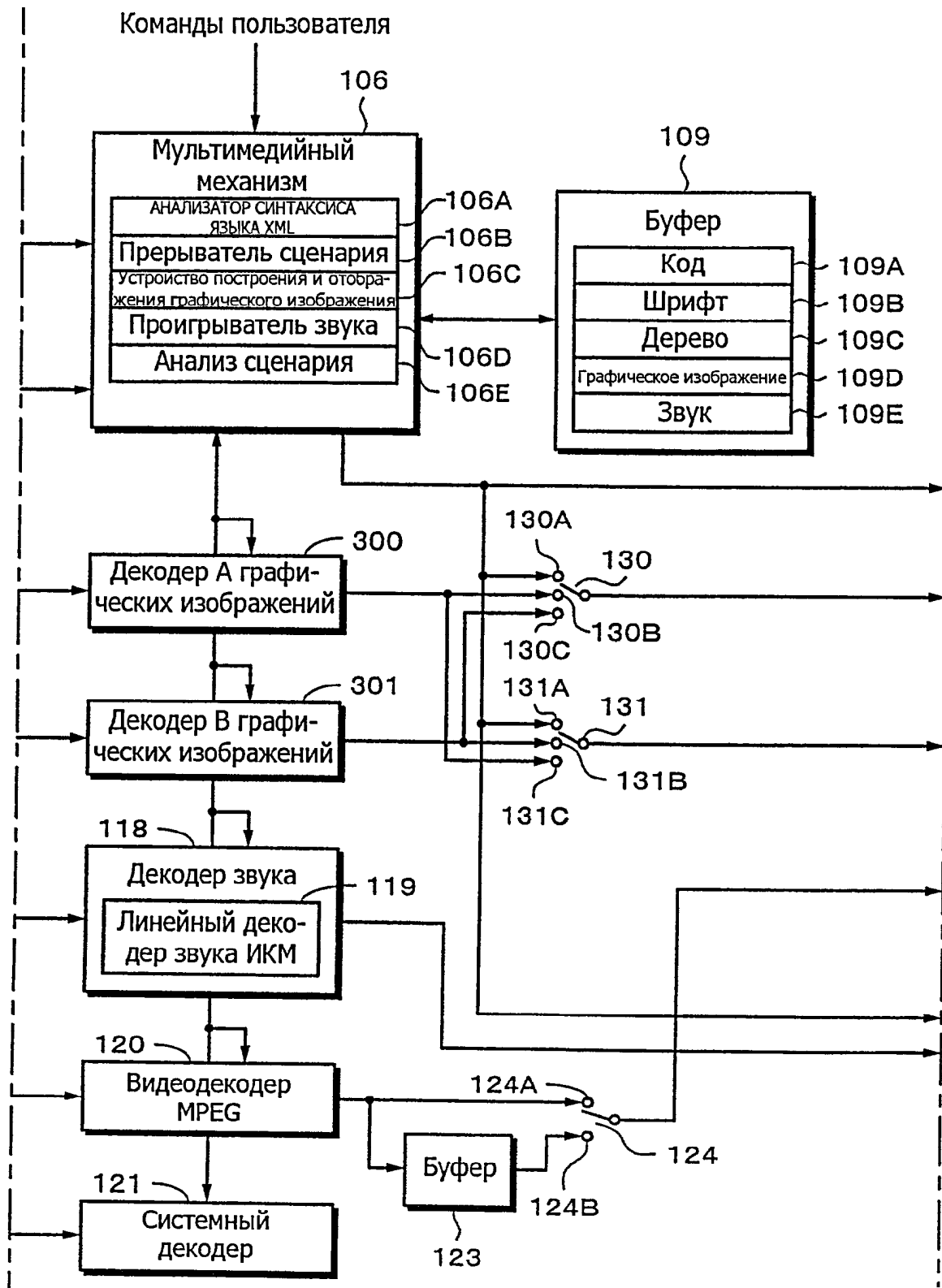
Фиг. 65В

Фиг. 66А | Фиг. 66В | Фиг. 66С

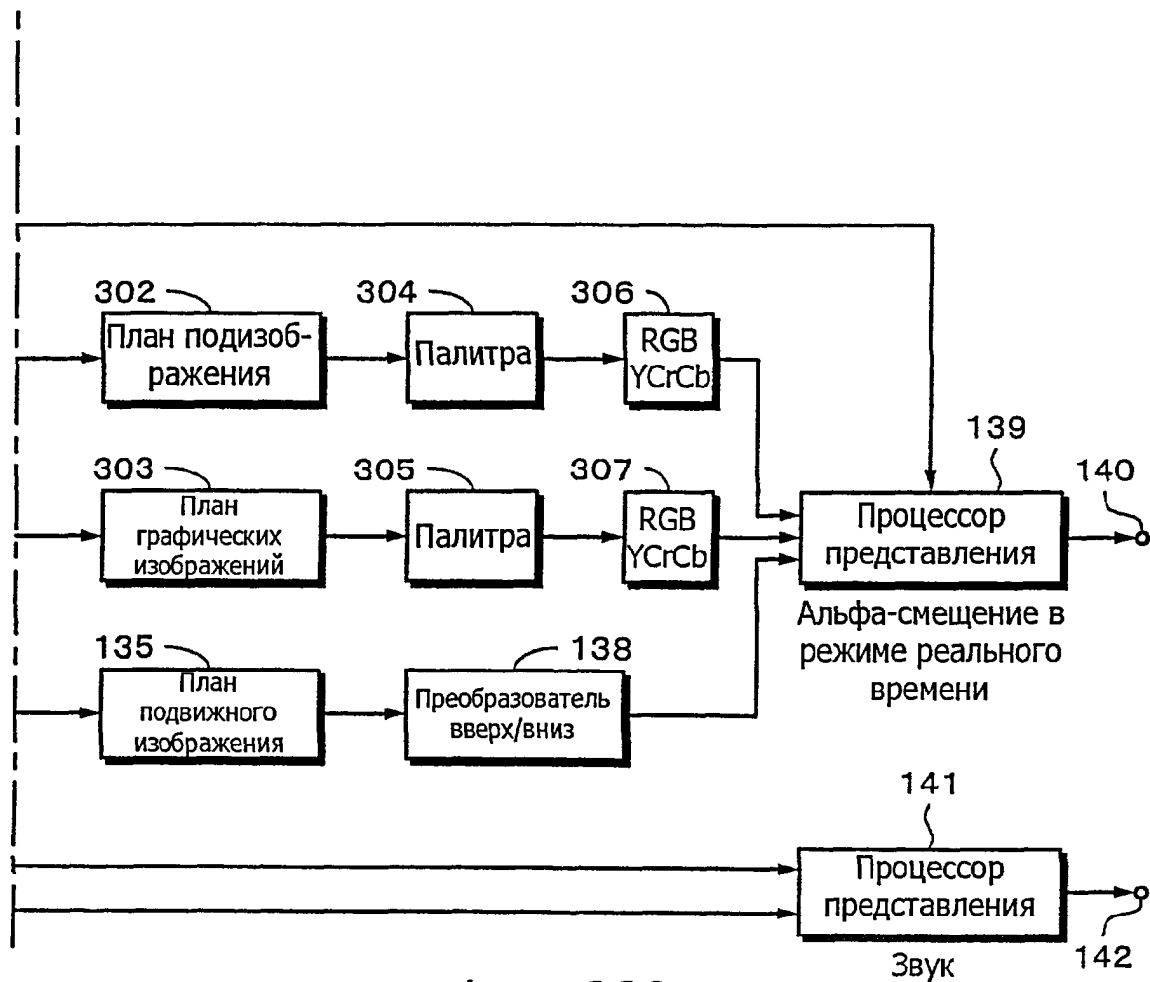
Фиг. 66



Фиг. 66А



Фиг. 66В



Фиг. 66С

## Описание номеров ссылок

10	План подвижного изображения
11	План подзаголовка
12	План графических изображений
22	Палитра
30	Виртуальное устройство воспроизведения ДГ
31	Команды устройства воспроизведения
32	Общий параметр
40	Программа управления воспроизведением
41	Способ
60	Экран меню
70	Сценарий
73А – 73М	Список воспроизведения
100	Декодер устройства воспроизведения
104	Буфер кода
105	Буфер содержания
106	Мультимедийный механизм
109	Буфер
110	Фильтр ИДП
116	Декодер подизображения
117	Декодер подвижного изображения
118	Декодер звука
120	Видеодекодер MPEG
132	План подизображения
133	План графического изображения

134	План неподвижного изображения
135	План подвижного изображения
226	Буфер декодера ПСГ
227	Декодер ПСГ
228	Буфер объекта
229	Буфер плана
231	Смеситель звука
500	Графический декодер А
501	Графический декодер В
502	План подизображения
503	План графических изображений
601A, 601B, 601C, 601D, 601E	протяженность